
AnalystGuide

Выпуск 0.0.4

февр. 03, 2026

Содержание

| | |
|--|-----------|
| 1 Предисловие | 2 |
| 1.1 Назначение | 2 |
| 1.2 Аудитория | 2 |
| 1.3 Общее описание курса | 2 |
| 1.4 Настройки, необходимые для прохождения курса | 3 |
| 2 Общие сведения по схемам процессов | 3 |
| 2.1 Общее описание | 3 |
| 2.2 Закладки схем бизнес-процессов | 3 |
| 2.3 Визуальный редактор | 5 |
| 3 Общие сведения по задачам | 6 |
| 3.1 Общее описание | 6 |
| 4 Общие сведения по ЭП | 6 |
| 4.1 Общее описание | 6 |
| 5 Замещения | 7 |
| 5.1 Общее описание | 7 |
| 6 Урок 1. Настройка схем и процедур бизнес-процессов | 8 |
| 6.1 Задание 1. Создание элементарной схемы бизнес-процесса | 8 |
| 6.2 Задание 2. Настройка субъектов | 12 |
| 6.3 Задание 3. Настройка процедур для схемы бизнес-процесса | 14 |
| 7 Урок 2. Детальная настройка схем бизнес-процессов | 18 |
| 7.1 Задание 1. Настройка ЭП и областей подписания | 18 |
| 7.2 Задание 2. Выдача ЭП пользователю | 20 |
| 7.3 Задание 3. Настройка требования ЭП на схеме процесса | 21 |
| 7.4 Задание 4. Дополнительная настройка шаблонов задач, субъектов и результатов процесса | 23 |
| 7.5 Задание 5. Работа с замещениями и перенаправлением задач | 24 |
| 7.6 Задание 6. Анализ запущенных процессов | 25 |
| 8 Самостоятельные задания | 26 |
| 8.1 Задание 1 | 26 |
| 8.2 Задание 2 | 27 |

| | |
|---|-----------|
| 9 Практические советы | 27 |
| 9.1 Входные параметры процедур | 27 |
| 9.2 Зарезервированные слова в jexl-скриптах процедур | 28 |
| 9.3 Список тэгов, доступных в шаблонах задачи | 28 |
| 9.4 Подходы для получения данных для обработки внутри процедуры | 28 |
| 9.5 Случаи, когда в субъекте несколько пользователей | 34 |
| 9.6 Завершение задачи при определенных условиях автоматически | 38 |

1 Предисловие

1.1 Назначение

Руководство по настройке WorkFlow в Системе Global.

1.2 Аудитория

Руководство предназначено для:

- Аналитиков Системы
- Администраторов Системы

1.3 Общее описание курса

В ходе данного курса слушатель познакомится с универсальным инструментом системы Global — схемами бизнес-процессов, которые можно подключать к объектам системы для их электронного согласования и автоматического изменения в ходе прохождения маршрута электронного документооборота. В рамках урока будет рассмотрено создание схем бизнес-процессов, их настройка для автоматизации таких задач, как определение перечня согласующих лиц и изменение состояний обрабатываемого документа.

Далее слушатель познакомится с детальными настройками схем бизнес-процессов, позволяющими указывать дополнительные требования к субъектам процесса (уточнение пользователей в ходе процесса согласования), к результатам выполнения задач (требование ЭП, комментария и прикрепления документа). Также слушатель познакомится с функционалом замещений, со способами анализа уже запущенных процессов и порядком работы со входящими задачами.

Полученные теоретические знания слушатель сможет закрепить в процессе выполнения самостоятельных практических занятий.

1.4 Настройки, необходимые для прохождения курса

Для выполнения самостоятельных практических заданий по уроку пользователю должен быть выдан доступ к приложениям «Настройка системы», «Управление бизнес-процессами» и «Управление закупками и складом».

2 Общие сведения по схемам процессов

2.1 Общее описание

Использующиеся в Системе схемы бизнес-процессов состоят из последовательностей выполняемых этапов и соединяющих эти этапы переходов, называемых «Состояния» и «Потоки» соответственно. Состояния связаны с пользователями, документами и действиями, в системе рассматриваемыми как «Субъекты», «Задачи» и «Результаты». Разметка потоков определяет порядок и условия прохождения состояний процесса. Схема позволяет настраивать заранее определенные маршруты автоматического создания задач для согласования документов, уведомления пользователей при наступлении определенных событий. Справочник схем бизнес-процессов в системе доступен в приложении «Управление бизнес процессами», меню «Справочники» -> «Схема бизнес-процесса», а также в приложении «Документооборот», меню «Настройки» -> «Настройки процессов» -> «Схемы бизнес-процессов».

2.2 Закладки схем бизнес-процессов

Состояния

Состояния – это этапы бизнес-процесса. Настроенное состояние позволяет создавать задачи для пользователей, выполнять процедуры, определять порядок и условия старта следующего этапа. Каждое состояние обладает наименованием, порядковым номером и типом. Любая схема по умолчанию имеет три встроенных состояния «Начало процесса», «Окончание процесса», «Аннулирование процесса» с установленным одноименным типом. Процесс всегда начинается с этапа «Начало процесса» и заканчивается на этапе «Окончание процесса», этап «Аннулирование процесса» происходит только при аннулировании процесса.

The screenshot shows the 'States' configuration screen with the following details:

Top Bar: Includes standard window controls (Minimize, Maximize, Close), a title bar 'DemandMH Направление - Согласование требований МСЧ', a status bar 'Текущий пользователь: [empty]', 'Дата версии: 16.12.2019', and a dropdown 'Состояние версии: Действующая'.

Main Area: A table titled 'Состояния' (States) with columns: Тип состояния БП (Type of state), Порядковый номер (Order number), Имя (Name), Условие пропуска (Skip condition), Отображать как... (Display as...), Контролируется (Controlled by), Субъект процесса (Process subject), Схема для поддер... (Scheme for sub...), Срок выполнения (Execution deadline), Вид срока выполн... (Type of execution deadline), Всегда показывать (Always show), При принятии не и... (On acceptance not), Вс (All), Иск. осн. (Searchable), Иск. неиск. (Non-searchable).

Data: The table contains several rows:

- Начало процесса:** Type: 1. Анулирование..., Order: 1, Name: Начало процесса, Skip condition: checked, Display as: checked, Controlled by: checked, Subject: Руководитель..., Scheme: Инициатор, Deadline: 10, Type: 10, Always show: checked, On acceptance not: checked, All: checked, Searchable: checked, Non-searchable: checked.
- Задание:** Type: 20 Установка шифро..., Order: 10, Name: Руководитель..., Skip condition: checked, Display as: checked, Controlled by: checked, Subject: 406/426-отдел, Scheme: Инициатор, Deadline: 20, Type: 20, Always show: checked, On acceptance not: checked, All: checked, Searchable: checked, Non-searchable: checked.
- Небольшое задание:** Type: 30 Требование..., Order: 20, Name: Инициатор, Skip condition: checked, Display as: checked, Controlled by: checked, Subject: Инициатор, Scheme: Инициатор, Deadline: 9999, Type: 9999, Always show: checked, On acceptance not: checked, All: checked, Searchable: checked, Non-searchable: checked.
- Небольшое задание:** Type: 40 Требование..., Order: 30, Name: Инициатор, Skip condition: checked, Display as: checked, Controlled by: checked, Subject: Инициатор, Scheme: Инициатор, Deadline: 9999, Type: 9999, Always show: checked, On acceptance not: checked, All: checked, Searchable: checked, Non-searchable: checked.
- Окончание процесса:** Type: 9999 Окончание..., Order: 9999, Name: Инициатор, Skip condition: checked, Display as: checked, Controlled by: checked, Subject: Инициатор, Scheme: Инициатор, Deadline: 30,40, Type: 30,40, Always show: checked, On acceptance not: checked, All: checked, Searchable: checked, Non-searchable: checked.

Bottom Tab: Includes tabs: Исходящие потоки (Outgoing flows), Входящие потоки (Incoming flows), Процедуры (Procedures), ЖЦ документа на состоянии (Lifecycle document for state), Области подписания (Signature areas).

Потоки

Настройка потоков регулирует последовательность прохождения этапов процесса, а также позволяет определять условия пропуска или активации этапов в зависимости от действий пользователя и параметров процесса.

Процедуры

Процедуры процесса – предустановленные системные методы и JEXL-скрипты, исполняемые при наступлении заданных условий для формирования сообщений пользователю, заполнения полей документов, проверки сложных условий, изменения состояний объектов системы и выполнения прочих автоматических сценариев.

Версионность

Вкладка «Версии» содержит список всех версий схемы бизнес-процесса, их состояния и описания. Активной считается версия в состоянии «Действующая». Если таких версий несколько, то выбрана будет версия с «Датой начала действия» ближайшей к текущей дате, но не превышающей ее. Доступно копирование предыдущих версий схемы для внесения изменений. Запущенный процесс будет следовать той версии, от которой он был запущен, даже если состояние версии будет изменено или созданы более актуальные версии.

Применимость к типам документов

Вкладка «Применимость к типам документов» содержит настраиваемый список типов объектов, для которых данная схема маршрута будет доступна. В случае соответствия типу объекта единственной схемы бизнес-процесса схема будет выбрана по умолчанию.

Субъекты процесса

Вкладка «Субъекты» содержит список субъектов схемы процесса. Субъект – это именованная группа пользователей системы, которая впоследствии доступна к обращению из схемы. Группа пользователей может быть задана как вручную прямо в схеме, так и заполняться в процессе исполнения процесса. По умолчанию любая схема имеет встроенный не изменяемый субъект «Инициатор». Этот субъект автоматически заполняется при старте процесса именем пользователя, который запускает процесс. Вкладка детализации «Настройки уточнения получателей» позволяет задать уточняющие параметры получателей.

Переменные

Вкладка «Переменные» содержит настраиваемый перечень переменных, используемых в ходе выполнения процесса. При запуске процесса и после выполнения стартовых процедур определяются их начальные значения, которые могут изменяться при прохождении этапов схемы. Значения переменных оцениваются в условиях пропуска/входа в поток, позволяя пропускать состояния в зависимости от предшествующих действий пользователей.

Результаты

Результаты в схеме бизнес-процесса — это перечень возможных операций пользователя, завершающих задачу. Действие пользователя с задачей — выбор им одного из предложенных результатов.

Процессы

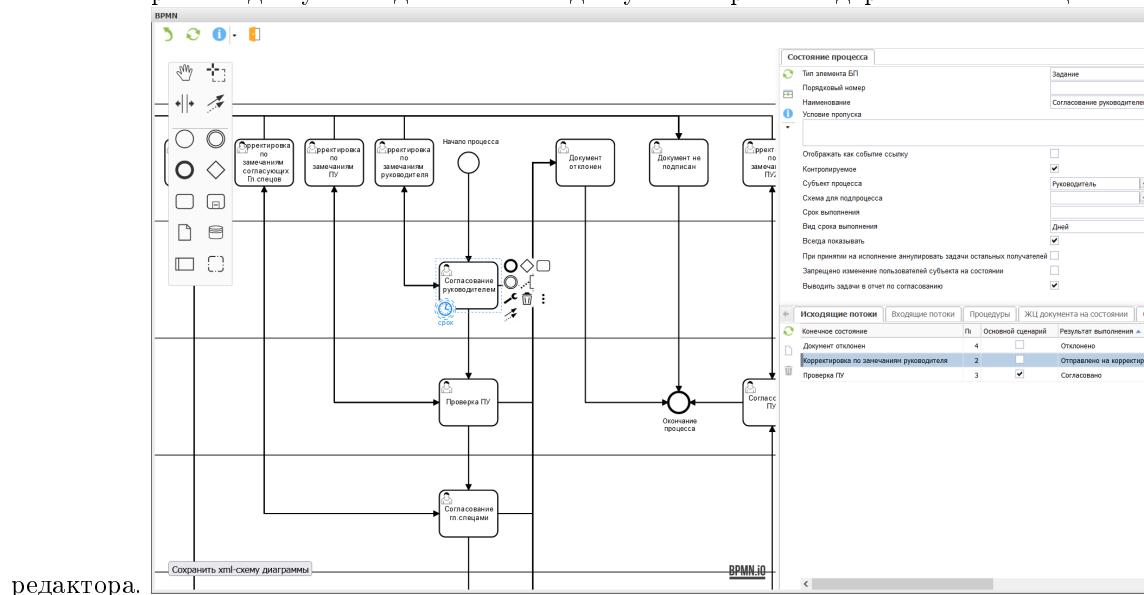
На вкладке «Процессы» представлен список всех запущенных процессов по данной схеме с возможностью создания новых и открытия карточек существующих.

Шаблоны задач

Вкладка «Шаблоны задач» предназначена для создания и управления шаблонами задач для текущей схемы. Для каждого шаблона указывается тип, процедура формирования текста, код, наименование, текст задачи. Использование процедур формирования текста позволяет, в частности, в тексте задачи отображать номер и дату электронного документа, прикрепленного к задаче. Шаблон задачи далее можно установить для определенного потока. При получении задачи получателем процесса текст задачи будет соответствовать указанному для потока шаблону.

2.3 Визуальный редактор

Система позволяет редактировать схемы бизнес-процессов с помощью встроенного визуального редактора. Внесенные в визуальную модель изменения отражаются в табличном представлении схемы, что позволяет настраивать схемы бизнес-процессов с помощью системы обозначений BPMN. Действительно и обратное: для уже созданной схемы доступно открытие и доработка с помощью визуального



редактора.

3 Общие сведения по задачам

3.1 Общее описание

Взаимодействие пользователей с объектами электронного документооборота осуществляется посредством обработки входящих и созданием исходящих задач. При помощи входящих задач происходит уведомление пользователей о наступлении событий и необходимости ознакомления, согласования или утверждения документов ТЭД. Путем создания исходящих задач формируются поручения на выполнение заданий. Каждая задача имеет инициатора и получателя(ей), тему, текстовое содержание, а также перечень прикрепленных документов. У каждого пользователя могут быть как исходящие, так и входящие задачи. Входящие задачи – это задачи, в которых пользователь является исполнителем в соответствии с установленным бизнес-процессом или сформированным другим сотрудником произвольным маршрутом. Исходящие задачи – это задачи, в которых пользователь является инициатором или отправителем. Задачи могут быть созданы пользователем вручную из списка задач. Также задачи могут формироваться автоматизировано на основании заданного для типа документа списка рассылки. Доступна возможность направления документа по произвольному маршруту на согласование, ознакомление или утверждение конкретным пользователям. Инициатором в этом случае будет являться пользователь, запустивший маршрут, а исполнителями по задачам будут назначенные лица.

Перенаправление задач

При выполнении от входящей задачи операции перенаправления новым ее исполнителем считается тот сотрудник, на которого задача перенаправлена. Задача более не показывается в списке входящих задач первоначального исполнителя и выполнить ее не может.

Создание подзадач

При выполнении от входящей задачи операции создания подзадач исходная задача остается за первоначальным исполнителем, а задачи другим сотрудникам создаются в рамках подпроцессов. После того, как исполнители выполняют свои задачи, исполнитель исходной задачи получает уведомление, от которого может перейти к исходной задаче и выполнить ее.

4 Общие сведения по ЭП

4.1 Общее описание

Электронная подпись — атрибут электронного документа системы, однозначно определяющий подлинность документа путем сравнения результата одностороннего преобразования значимых реквизитов документа (либо всего двоичного файла в случае подписания электронного документа) с предварительно сохраненными результатами аналогичного преобразования. В случае изменения участвующих в подписи реквизитов будет получен отличающийся результат, что лишит документ силы. В Системе электронная подпись используется для заверения выполнения пользователем таких задач как подтверждение, согласование или ознакомление с документом.

Виды ЭП

Доступны три умолчательных вида ЭП: ознакомительная, согласующая и утверждающая. Также в некоторых случаях доступно выполнение различных механик, запускаемых на подписание электронной подписью одного из видов, например, регистрация документа при его подписании утверждающей ЭП.

Области подписания

Возможна следующая ситуация: Инициатор запускает на согласование документ без заполненной даты утверждения и подписывает его ЭП. Руководитель Утверждает документ, заполняет дату утверждения и подписывает своей подписью. В таком случае проверка ЭП первого сотрудника покажет несоответствие подписи (так как один из атрибутов изменился), хотя введенные им данные не менялись. Для обработки подобных ситуаций реализовано разделение атрибутов по областям подписания: в ЭП каждого сотрудника указывается, какие атрибуты он подписывает.

5 Замещения

5.1 Общее описание

В системе доступна настройка замещения пользователей на требуемый период времени. Возможно одновременное замещение пользователем неограниченного количества других сотрудников. Замещение пользователя осуществляется посредством внесения записи в список замен, обозначающей соответствие между замещаемым и замещающим пользователями, а также обозначает временной период, в течение которого замещение действительно.

На период действия замещения замещающий пользователь получает следующие права замещаемого пользователя:

Доступ к документам

Замещающему пользователю доступен просмотр, создание и изменение документов не только со своими правами, но и одновременно с правами замещаемого пользователя (по принципу объединения прав).

Обработка задач

В перечне входящих и исходящих задач замещающего пользователя отображены не только свои задачи, но и задачи замещаемого пользователя. Доступна обработка и выполнение входящих задач замещаемого пользователя (при этом в системе фиксируется факт того, что задача была выполнена замещающим пользователем), редактирование исходящих задач.

Подписание документов

Доступно согласование и подписание ЭП документов, назначенных замещаемому пользователю (при этом в системе фиксируется факт того, что документ подписан или согласован замещающим пользователем).

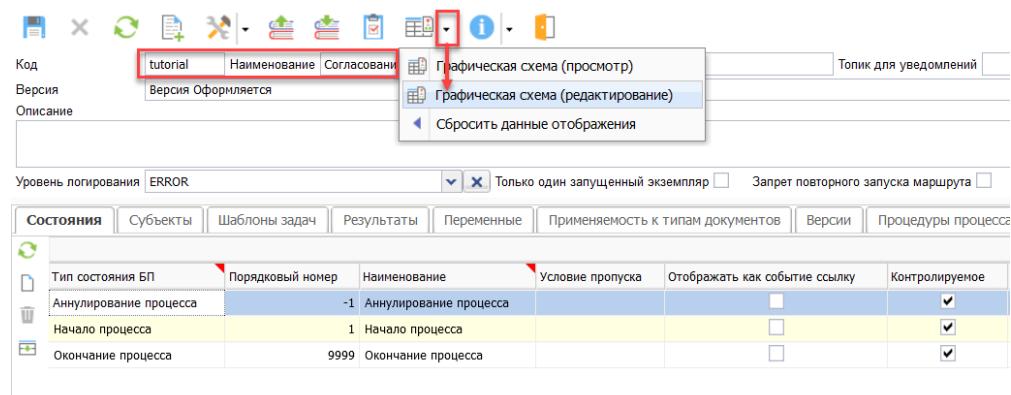
6 Урок 1. Настройка схем и процедур бизнес-процессов

6.1 Задание 1. Создание элементарной схемы бизнес-процесса

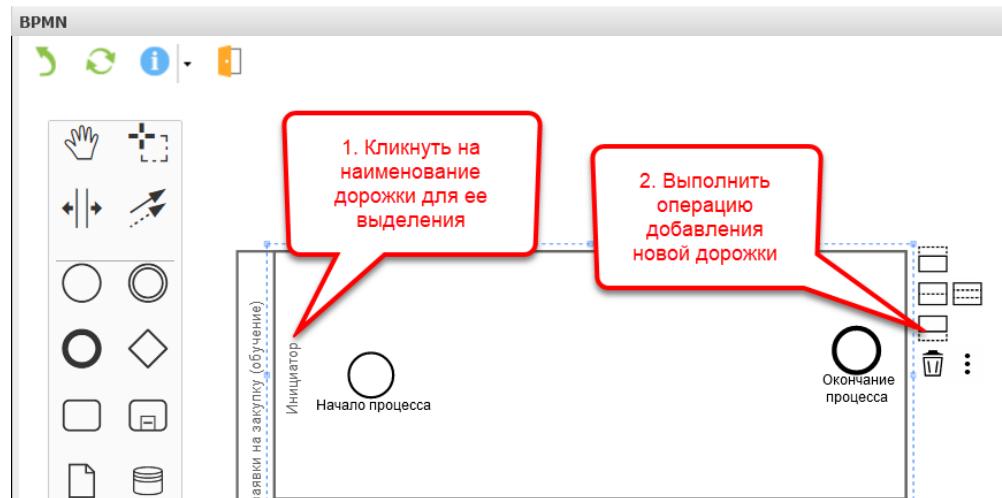
Создайте схему процесса с одним этапом согласования. Присоедините схему к типу объекта.

Для выполнения задания необходимо:

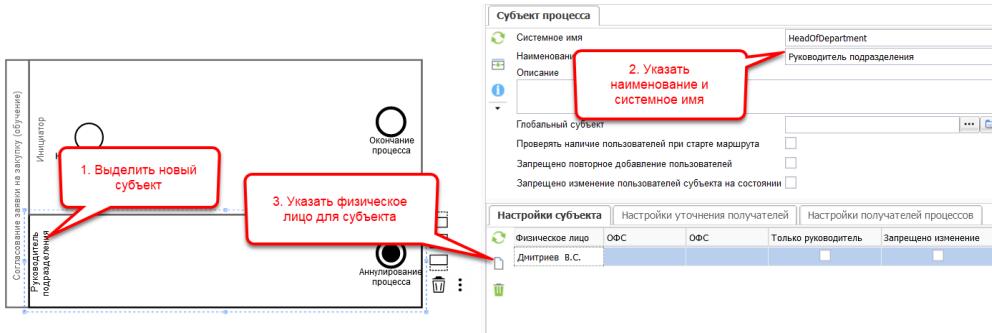
- Создать новую схему процесса (Управление бизнес-процессами -> Справочники -> Схема бизнес-процесса). Указать ее код и наименование, перейти в режим редактирования графической схемы.



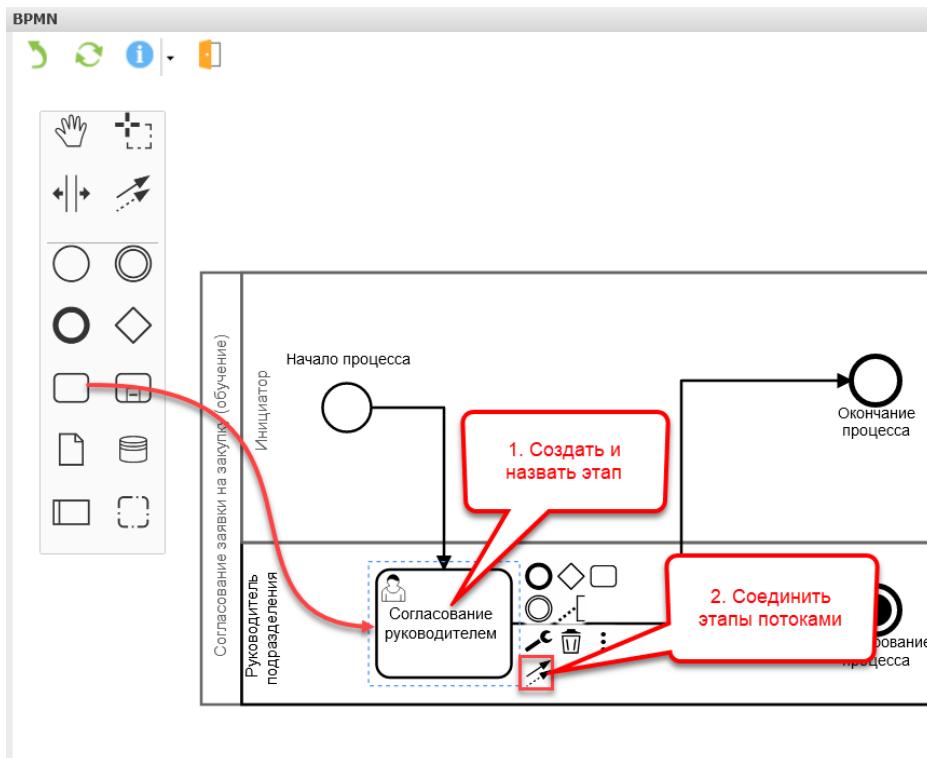
- Выделить дорожку «Инициатор» и выполнить появившуюся операцию добавления новой дорожки.



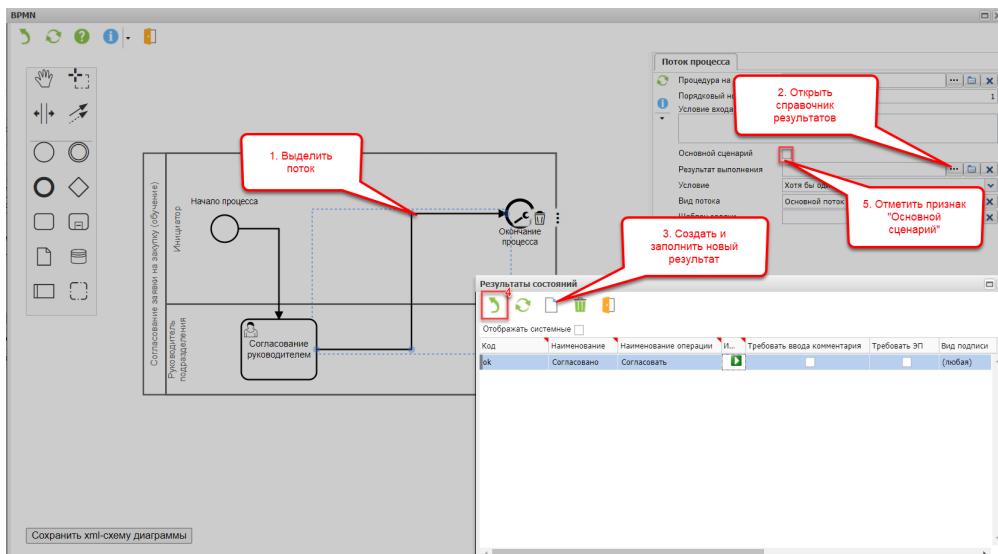
- Выделить созданную дорожку, указать ей системное имя и наименование. На закладке «Настройки субъекта» создать строку и указать физ.лицо. В данном задании для упрощения предлагается выбрать конкретного сотрудника, в последующих заданиях будет осуществлена более сложная динамическая настройка субъекта.



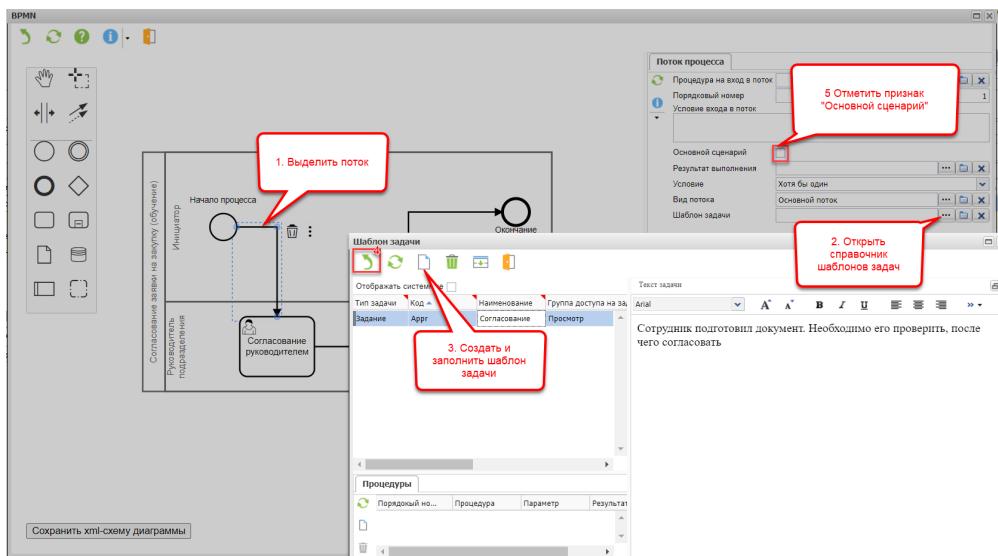
- Перетащить с панели инструментов на дорожку «Руководитель» блок с типом «Задание», назвать его. Выделив новый блок, выбрать операцию создания потоков и протянуть стрелку к этапу «Окончание процесса». Аналогичным образом направить поток из состояния «Начало процесса» в состояние «Согласование руководителем»



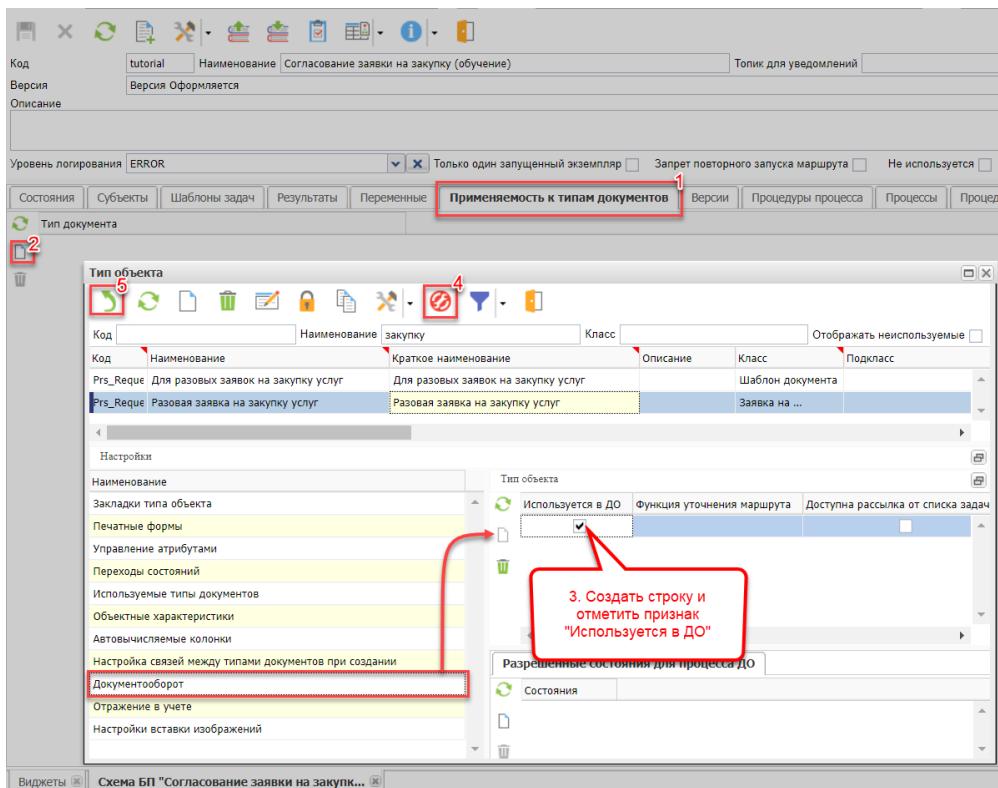
- Выделить поток, ведущий от этапа согласования к окончанию процесса, и указать результат (кнопка, которую в своей задаче нажмет пользователь), требуемый для его активации. Отметить признак «Основной сценарий» на потоках, которые будут считаться умолчательными в случае корректного согласования.



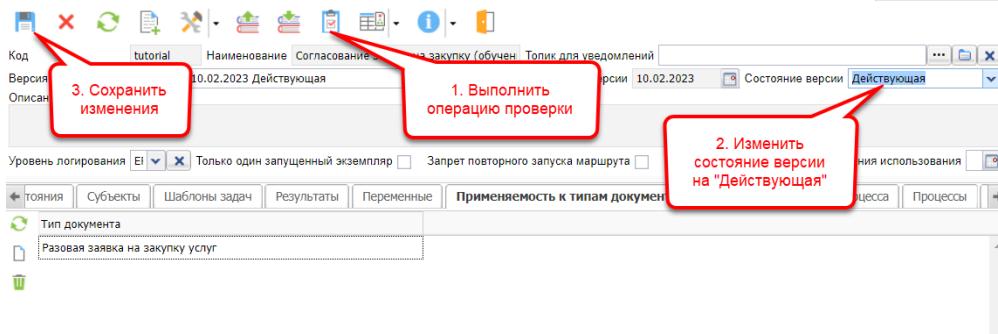
- Выделить поток, ведущий начала процесса к этапу согласования, и указать шаблон задачи, которая будет направлена пользователю при активации потока и переходе в следующее состояние. Отметить признак «Основной сценарий».



- Завершить работу визуального редактора операций «Выбор».
- Перейти на закладку «Применимость к типам документов» и выполнить операцию добавления. В открывшемся справочнике найти требуемый тип объекта. В детализации к типу объекта перейти на закладку «Документооборот» и убедиться в том, что существует строка с отмеченным признаком «Используется в ДО». Если такой строки нет, создать ее, отметить признак и применить изменения операцией «Сбросить кэш настройки». Подтвердить выбор типа объекта операцией «Выбор».



- Проверить схему операцией проверки. На выполнение операции в нижней части экрана появится лог проверки, в котором будет указано, корректно ли соединены состояния, для всех ли потоков указаны требуемые шаблоны задач и результаты. Изменить состояние версии на «Действующая» и сохранить изменения.



- Чтобы пользователи могли видеть прогресс прохождения согласования документа, необходимо настроить отображение универсальной закладки «Маршруты». Для этого следует открыть справочник типов объектов (Настройки -> Настройки документации -> Типы объектов), найти требуемый. В разделе «Закладки типа объекта» найти закладку «Маршруты» и отметить ее активность. Сохранить изменения, применить изменения операцией «Сбросить кэш настройки».

- Для проверки открыть документ настроенного типа и запустить процесс документооборота по нему.
- Для проверки маршрута следует последовательно выполнить все задачи по нему. Доступны два способа:
 - Авторизоваться в системе от имени пользователя-исполнителя, выполнить задачу, открыв список входящих задач через меню Сервис -> Входящие задачи.
 - Перейти в карточку задачи с закладки «Маршрут» и выполнить ее от своего имени, пользуясь привилегиями супер-пользователя.

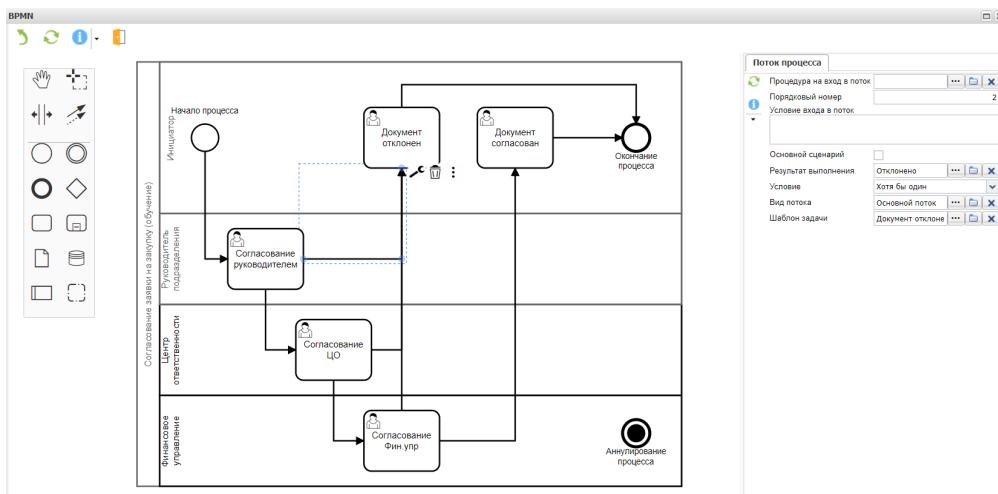
6.2 Задание 2. Настройка субъектов

Создайте новую версию схемы, предусматривающую согласование также Центром ответственности и Финансовым управлением, согласующие сотрудники которых будут определяться автоматически. Реализуйте возможность отклонения каждой из согласующих инстанций.

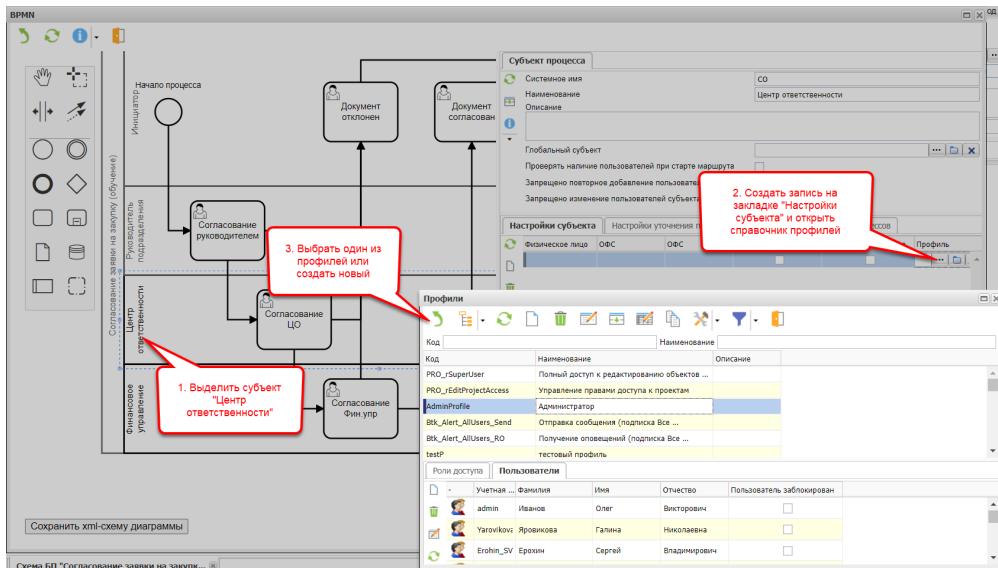
Для выполнения задания необходимо:

- Создать новую версию схемы процесса, скопировав предыдущую. Ввести дату начала ее действия и описать изменения. Переключиться на редактирование новой версии.

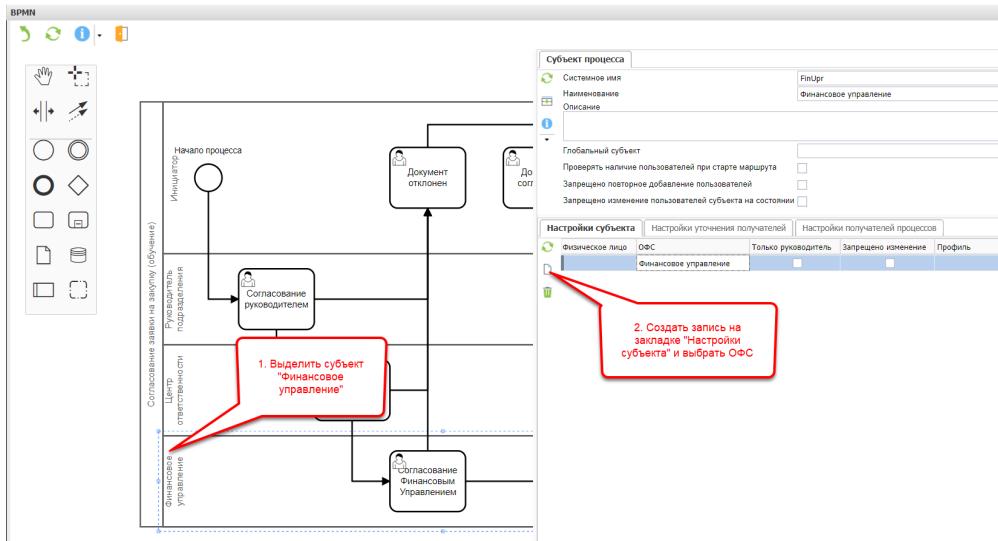
- С помощью визуального редактора добавить новых субъектов, новые состояния и переходы.



- Выделить субъект «Центр ответственности». На закладке «Настройки субъекта» создать строку и перейти в справочник профилей. Выбрать один из имеющихся профилей или создать новый и указать для него пользователей. В таком случае при активации состояния задачи по нему получат все пользователи, имеющие данный профиль. Если новый сотрудник получит данный профиль, он увидит и ранее направленные на него задачи.



- Выделить субъект «Финансовое управление». На закладке «Настройки субъекта» создать строку и выбрать ОФС из справочника. В таком случае при активации состояния задачи по нему получат все сотрудники, числящиеся в ОФС на момент отправки задач.

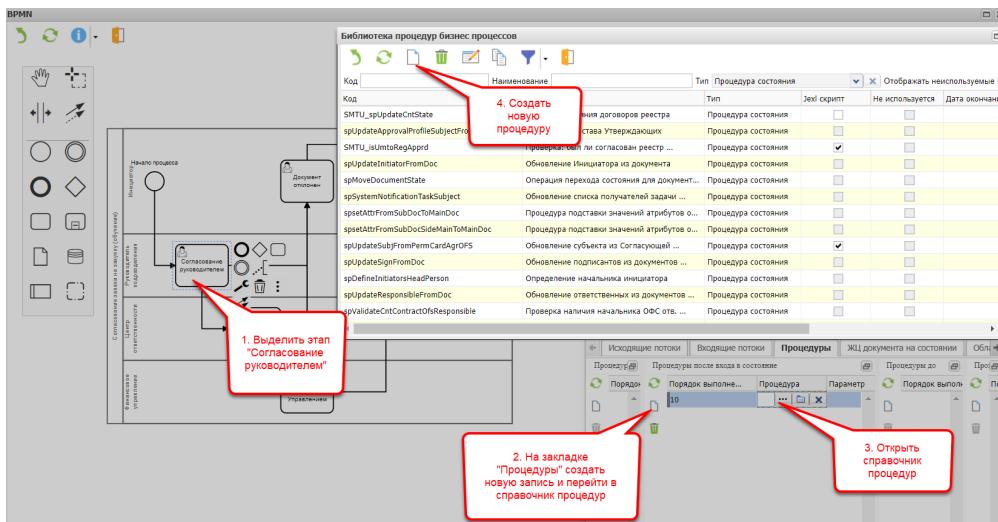


6.3 Задание 3. Настройка процедур для схемы бизнес-процесса

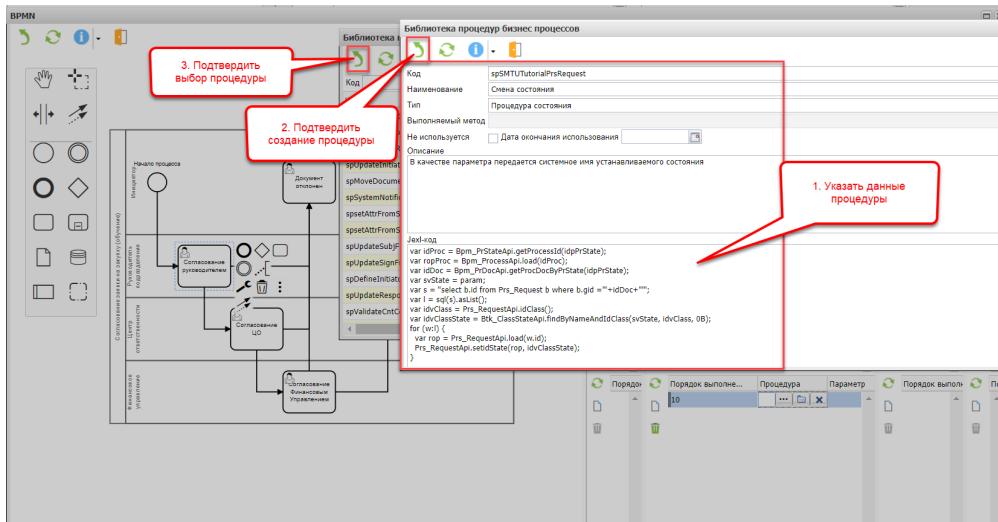
Реализуйте автоматическую смену состояния документа при его согласовании и отклонении. Определите руководителя инициатора процедуры на основании атрибута «Заявитель» документа.

Для выполнения задания необходимо:

- Выделить этап «Согласование руководителем». Перейти на закладку «Процедуры» и создать новую запись в разделе «Процедуры» после старта процесса. Перейти в справочник и создать новую процедуру.



- Указать код, наименование и содержание (см. следующий пункт) процедуры. Завершить создание процедуры кнопкой «Выбор», после чего в справочнике процедур появится только что созданная. Выбрать ее из справочника.



- Чтобы не писать процедуру с нуля, можно найти аналогичную и адаптировать ее под требуемый класс. Например, в системе уже доступна процедура «Перевод состояния лота». Для адаптации необходимо изменить в тексте процедуры все упоминания класса закупки (Prs_Lot) на Prs_Request:

```

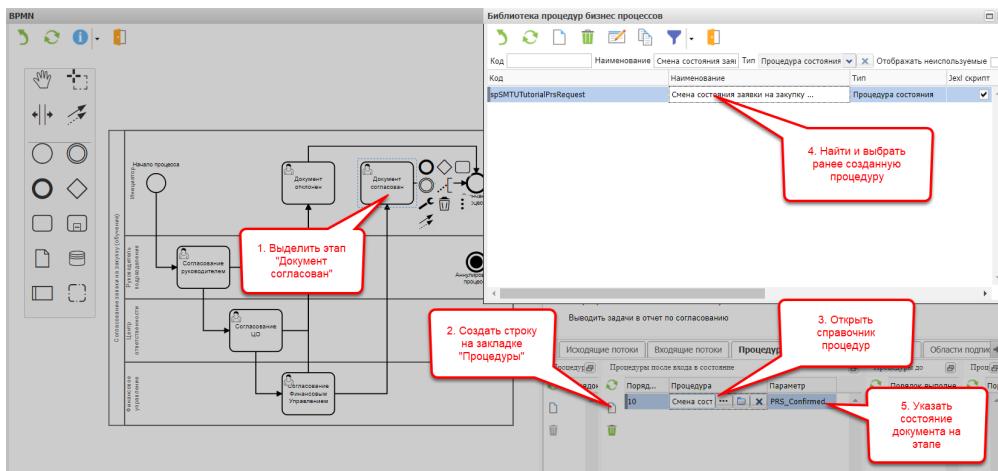
var idProc = Bpm_PrStateApi.getProcessId(idpPrState);
var ropProc = Bpm_ProcessApi.load(idProc);
var idDoc = Bpm_PrDocApi.getProcDocByPrState(idpPrState);
var svState = param;
var s = "select b.id from Prs_Lot b where b.gid ='"+idDoc+"'";
var l = sql(s).asList();
var idvClass = Prs_LotApi.idClass();
var idvClassState = Btk_ClassStateApi.findByNameAndIdClass(svState, idvClass, OB);
for (w:l) {
    var rop = Prs_LotApi.load(w.id);
    Prs_LotApi.setidState(rop, idvClassState);
}

```

- Найти в справочнике типов объекта требуемый и зафиксировать системные имена используемых состояний для того, чтобы использовать их в качестве параметров.

| Системное имя | Наименование |
|---------------|--------------|
| PRS_Annulled | Аннулировано |
| PRS_Forming | Формируется |
| PRS_Hold | Направлено |
| Coordinating | Соглашается |
| PRS_Confirmed | Утверждено |

- Указать состояние «Соглашается» (Coordinating) как параметр для процедуры на этапе «Согласование руководителем».
- Выделить этап «Документ согласован» и выбрать для него из справочника процедур только что созданную процедуру смены состояния. В качестве параметра указать состояние PRS_Confirmed.



- Аналогичным образом указать перевод состояния документа в «Аннулирован» при активации этапа «Документ отклонен».

Далее будет описано заполнение субъекта «Руководитель подразделения» автоматически на основании атрибутов документа. Для этого необходимо:

- Определить, на основании каких атрибутов документа определять пользователя. Документ содержит атрибут «Заявитель» — ссылка на ОФС. Запись ОФС содержит атрибут «Руководитель» — ссылка на требуемое физическое лицо.

The screenshot shows the 'Руководитель' form for document #78. It includes fields for 'Цель закупки' (Разовая заявка на потребность), 'Заявитель' (5100, СМТ-1), 'Центр ответственности' (Управление капитального строительства), 'Подразделение закупщика', 'Место хранения обеспечения' (ОФС №5100), and 'Шаблон документа' (Разовая заявка на потребность). A red arrow points from the 'Заявитель' field to the 'Сотрудники' tab of the 'Справочник' form, which lists employees with 'Иванов О.В.' selected.

- Написать SQL-запрос, который будет возвращать id физ.лица — получателя задачи — на основании gid исходной заявки на потребность:

```
select coalesce (p.iduser,0) as iduser
from Prs_Request r
join bs_ofstructure o on o.id = r.idofstructure
join bs_employee e on e.id = o.idemployee
join bs_person p on e.idperson=p.id
where r.gid = '72951/1706'
```

- Найти аналогичную процедуру «Заполнение куратора договора»:

```

var idProc = Bpm_PrStateApi.getProcessId(idpPrState);
var svSubj = param;
var idvDoc = Bpm_PrDocApi.getProcDocByPrState(idpPrState);
var ropProc = Bpm_ProcessApi.load(idProc);
var ropTargetOpt = Bpm_PrSubjectApi.getByPSSubject(ropProc, svSubj);
var ropTarget = ropTargetOpt.get();
var s = `select coalesce (u.iduser,0) as iduser
from cnt_contract r
join bs_employee e on r.IDCURATOREMPLOYEE=e.id
join bs_person u on e.idperson = u.id
where r.gid =  '` + idvDoc + `';
var l = sql(s).asList();
for (w:l) {
    if (w.iduser == 0) {raise("Внимание! Для договора не указан куратор.");}
    Bpm_PrSubjUserApi.register(ropTarget, w.iduser, 0B, 0B);
}

```

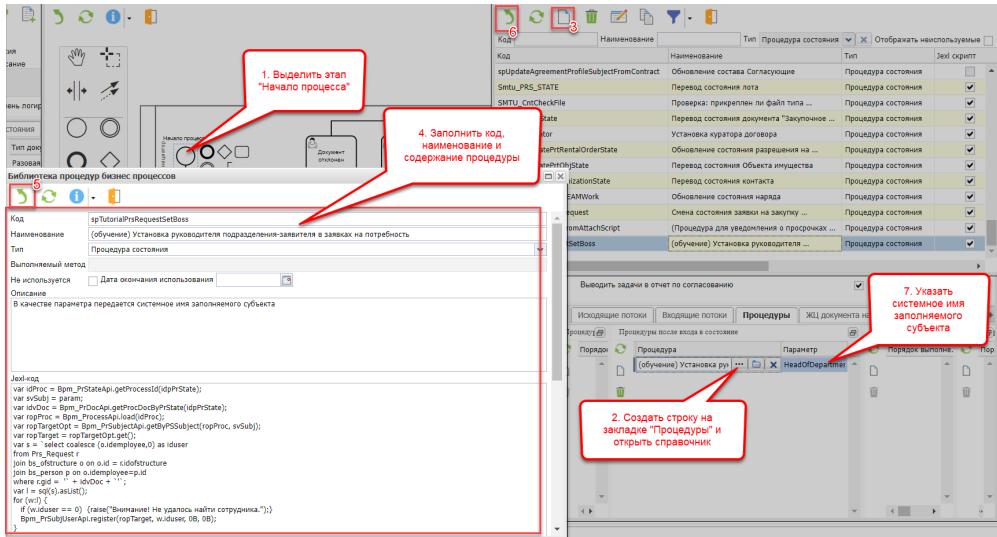
SQL запрос необходимо заменить на написанный ранее, чтобы получился следующий результат:

```

var idProc = Bpm_PrStateApi.getProcessId(idpPrState);
var svSubj = param;
var idvDoc = Bpm_PrDocApi.getProcDocByPrState(idpPrState);
var ropProc = Bpm_ProcessApi.load(idProc);
var ropTargetOpt = Bpm_PrSubjectApi.getByPSSubject(ropProc, svSubj);
var ropTarget = ropTargetOpt.get();
var s = `select coalesce (p.iduser,0) as iduser
from Prs_Request r
join bs_ofstructure o on o.id = r.idofstructure
join bs_employee e on e.id = o.idemployee
join bs_person p on e.idperson=p.id
where r.gid =  '` + idvDoc + `';
var l = sql(s).asList();
for (w:l) {
    if (w.iduser == 0) {raise("Внимание! Не удалось найти руководителя ОФС-заявителя.");}
    Bpm_PrSubjUserApi.register(ropTarget, w.iduser, 0B, 0B);
}

```

- Для состояния «Начало процесса» на закладке «Процедуры» создать строку, открыть справочник процедур, создать новую и вставить в нее JEXL-код. Подтвердить выбор. В качестве параметра для процедуры указать системное имя субъекта «Руководитель подразделения».



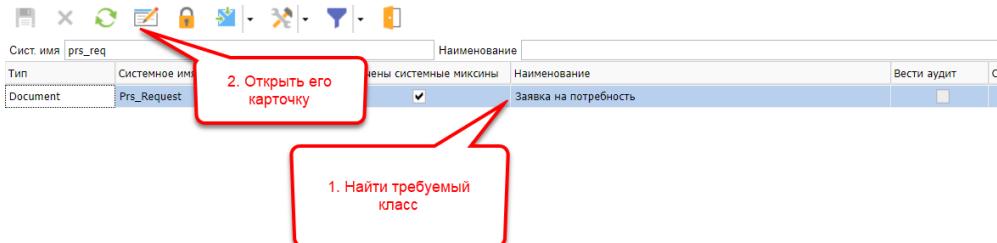
7 Урок 2. Детальная настройка схем бизнес-процессов

7.1 Задание 1. Настройка ЭП и областей подписания

Настройте возможность подписания документа «Заявка на потребность» (Prs_Request) электронной подписью. Настройте две разных области подписания: одну для запускаемого на согласование документа и одну для утвержденного документа, включающую дополнительные атрибуты.

Для выполнения задания необходимо:

- В приложении «Настройка системы» перейти в перечень классов системы (Сущности -> Классы) и найти требуемый (Prs_Request). Перейти в его карточку.



- В списке атрибутов класса отметить признаками те атрибуты, которые будут использоваться в электронной подписи.

Системное имя: Prs_Request
Наименование: Заявка на потребность

| Название атрибута | Тип атрибута | Значение по умолчанию | Ссылается на класс | Объектная характеристика | Хранится в json | Используется в подписи |
|------------------------|----------------------------|-----------------------|--------------------------|--------------------------|-----------------|------------------------|
| idClass | Long | Basic | | | | |
| sNumDocBMs_dz | Number | Basic | | | | |
| sNumDoc_dz | Varchar | Basic | | | | |
| sNumDecidVer_dz | Long | Basic | | | | |
| sRegNumberBM_dz | Number | Basic | | | | |
| sRegNumber_dz | Varchar | Basic | | | | |
| sRegNumberIdVer_dz | Long | Basic | | | | |
| dShortageAndLossesItem | Long | RefObject | Bs_ShortageAndLossesItem | | | |
| dDoc | Date | Basic | | | | |
| dRegDate | Date | Basic | sysdate | | | |
| dAccept | Date | Basic | | | | |
| dPryVer | Заказ | RefObject | Bs_PryVer | ✓ | | ✓ |
| id | Long | Basic | | | | |
| idSrcObj | Источник | RefAnyObject | Bk_Object | ✓ | | |
| idSrc | Источник | RefAnyObject | | | | |
| idDelPriority | Критичность поставки | RefObject | Prs_DelPriority | | | |
| idCostCenter | MBS | RefObject | Bs_CostCenter | ✓ | | |
| idMNGPerson | Менеджер | RefObject | Bs_MNGPerson | | | ✓ |
| idStockOut | Место хранения обеспечения | RefObject | Stk_Stock | ✓ | | |
| idPurchaseDirection | Направление закупок | RefObject | Bs_PurchaseDirection | | | |
| idNeedsDirection | Направление потребности | RefObject | Bs_NeedsDirection | ✓ | | |

- В справочнике бизнес-объектов (Сущности -> Бизнес-объекты) найти требуемый бизнес-объект. Добавить в умолчательную область подписания атрибуты.

Системное имя класса: Prs_Request
Наименование: Заявка на потребность

Области подписания

| Код | Наименование | Не используется | Дата окончания использования |
|---------|----------------------------------|-----------------|------------------------------|
| default | Умолчательная область подписания | | |

Только атрибуты текущего BO

| Сист. имя класса | Класс | Сист. имя атрибута | Атрибут | Бизнес-объект |
|------------------|-----------------------|--------------------|------------------|---------------|
| Prs_Request | Заявка на потребность | idPryVer | Заказ | Prs_Request |
| Prs_Request | Заявка на потребность | dRegDate | Дата регистрации | Prs_Request |
| Prs_Request | Заявка на потребность | idMNGPerson | Менеджер | Prs_Request |
| Prs_Request | Заявка на потребность | dDoc | Дата | Prs_Request |

- Создать новую область подписания и добавить в нее другой набор атрибутов (например, включающий дополнительные атрибуты).

Системное имя класса: Prs_Request
Наименование: Заявка на потребность

Области подписания

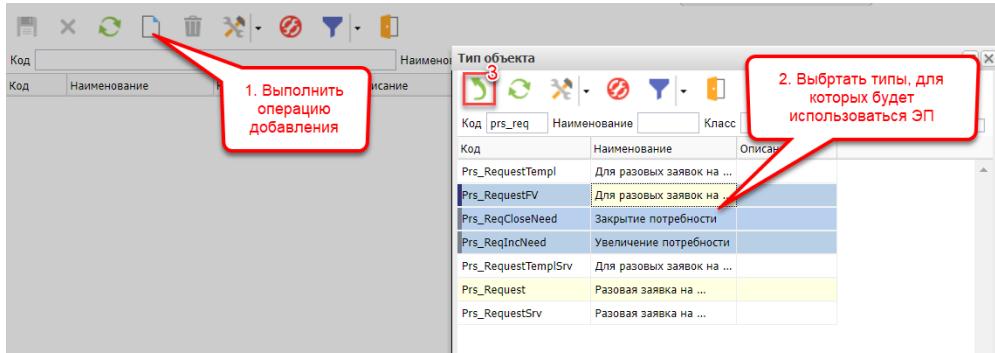
| Код | Наименование | Не используется | Дата окончания использования |
|---------|---------------------------------------|-----------------|------------------------------|
| default | Умолчательная область подписания | | |
| manager | (обучение) Подписание утвержденног... | | |

Только атрибуты текущего BO

| Сист. имя класса | Класс | Сист. имя атрибута | Атрибут | Бизнес-объект |
|------------------|-----------------------|--------------------|------------------|---------------|
| Prs_Request | Заявка на потребность | idMNGPerson | Менеджер | Prs_Request |
| Prs_Request | Заявка на потребность | dDoc | Дата | Prs_Request |
| Prs_Request | Заявка на потребность | dRegDate | Дата регистрации | Prs_Request |
| Prs_Request | Заявка на потребность | dAccept | Дата утверждения | Prs_Request |
| Prs_Request | Заявка на потребность | dPryVer | Заказ | Prs_Request |

- Перейти в список типов объектов, к которым применимы ЭП (Сервисы -> Настройка ЭП ->

Подписи на тип объекта) и добавить требуемые типы. Доступна более общая настройка по целым классам (Сервисы -> Настройка ЭП -> Подписи на класс)

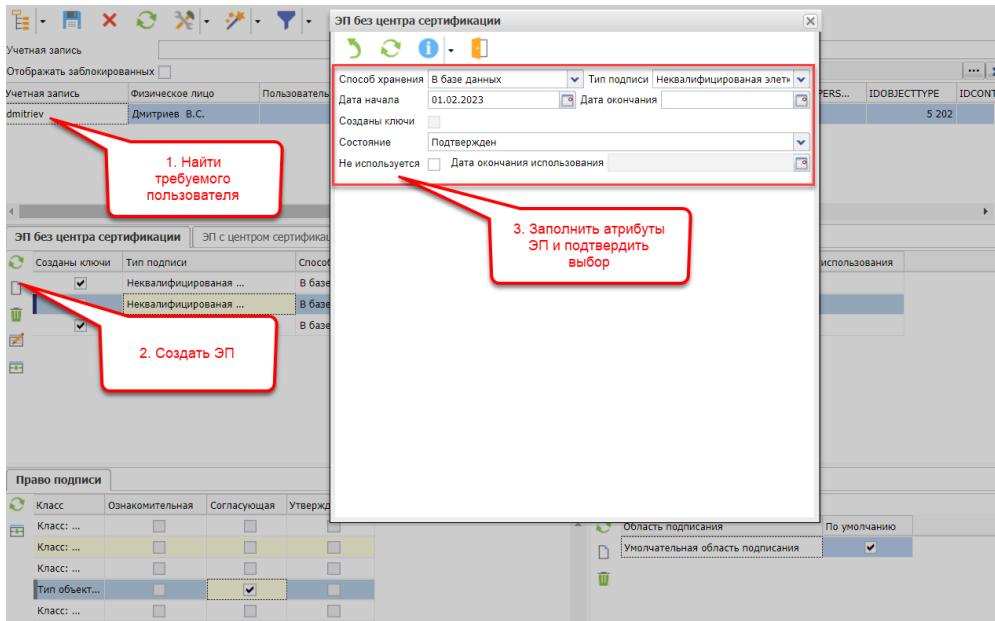


7.2 Задание 2. Выдача ЭП пользователю

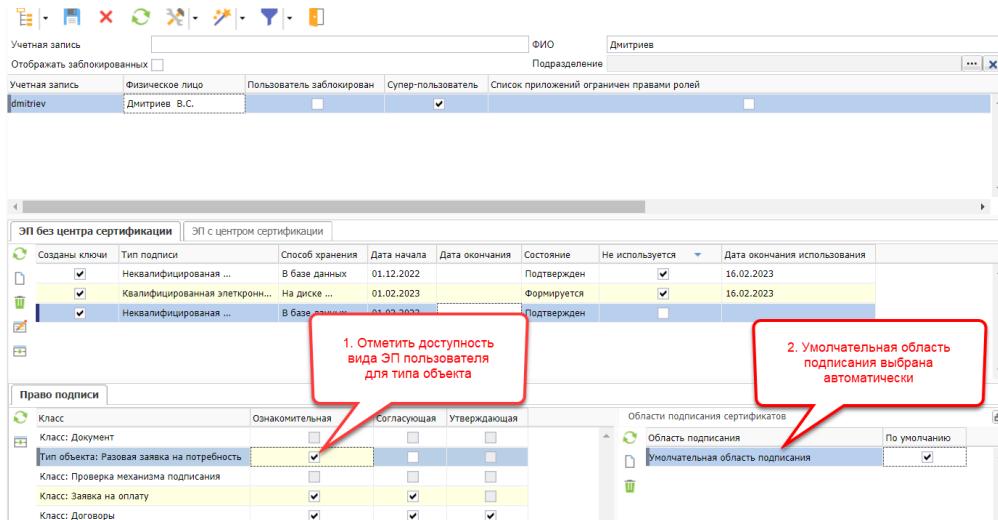
Выдайте электронные подписи с разными областями подписания своему пользователю и утверждающему.

Для выполнения задания необходимо:

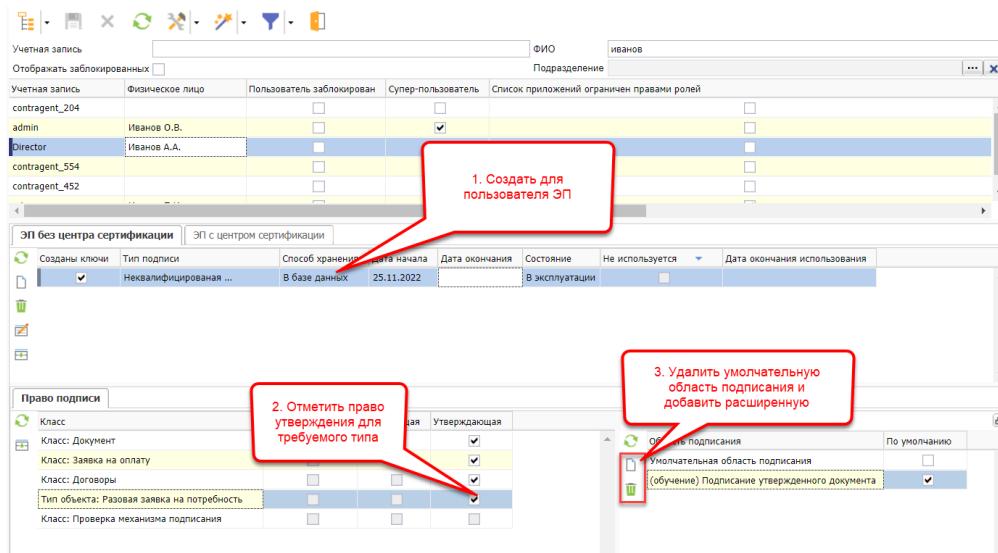
- Перейти в список пользовательских электронных подписей (Сервисы -> Настройка ЭП -> Настройка ЭП на пользователя), найти требуемого пользователя и выполнить для него операцию создания ЭП. Указать ее реквизиты и подтвердить выбор.



- Для созданной ЭП отметить доступность ознакомительного подписания для требуемого типа объекта. Умолчательная область подписания (в которую ранее было добавлено меньшее количество атрибутов) подобрана автоматически.



- Аналогичным образом для пользователя, который будет утверждать документ, создать ЭП. Отметить право утверждения для данной ЭП и выбрать ранее настроенную область подписания с большим набором атрибутов.

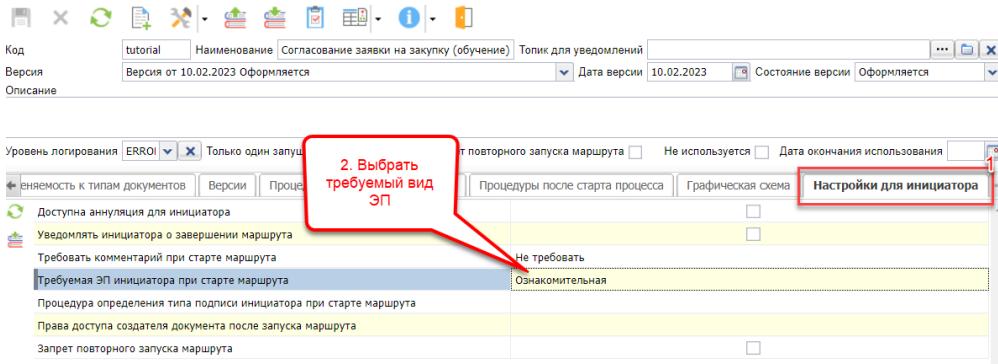


7.3 Задание 3. Настройка требования ЭП на схеме процесса

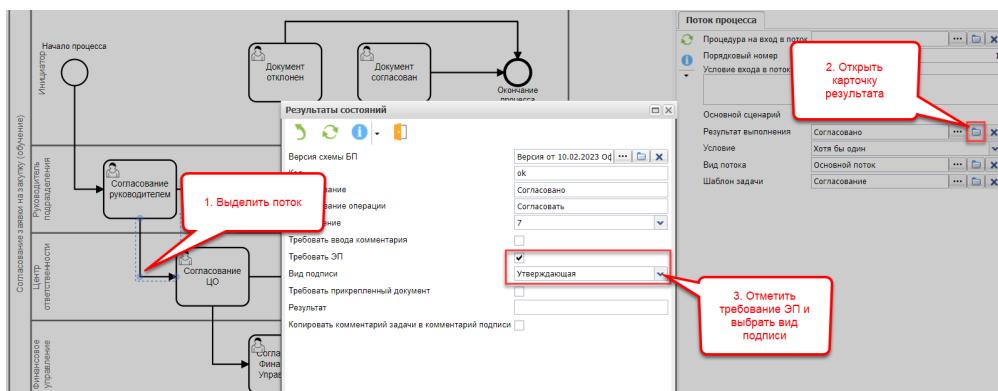
Настройте требование электронного подписания документа при старте маршрута и при его утверждении.

Для выполнения задания необходимо:

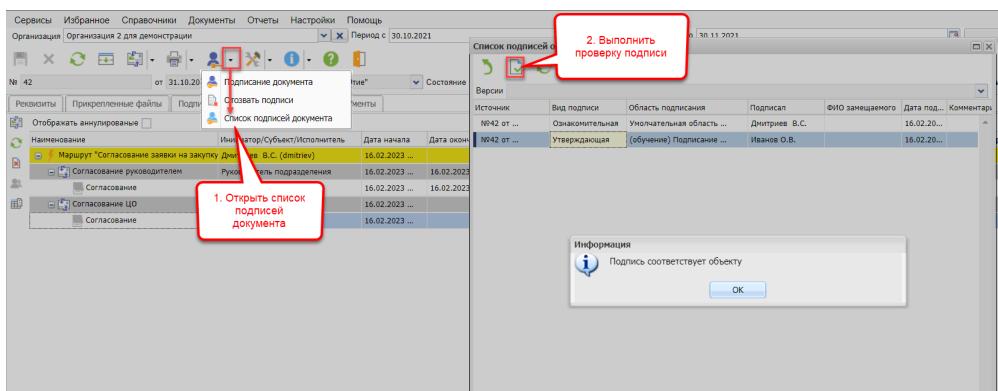
- Создать новую версию схемы процесса, скопировав предыдущую.
- В схеме процесса перейти на закладку «Настройки для инициатора» и указать требуемый вид ЭП для запуска маршрута.



- На графической схеме выбрать поток, для активации которого будет требоваться ЭП. Открыть карточку результата, требуемого для активации потока и выбрать в ней необходимость ЭП и требуемый её вид.



- Изменить состояние версии обратно на «Действующая», сохранить изменения.
- Открыть карточку одного из документов и запустить маршрут ЭД. Подтвердить согласие с подписанием документа в диалоговых окнах.
- Авторизоваться в системе от имени утверждающего сотрудника. Открыть список входящих задач (Сервис -> Входящие задачи) и выполнить операцию согласования.
- Для проверки подписей перейти в карточку документа, открыть список ЭП и выполнить соответствующую операцию.

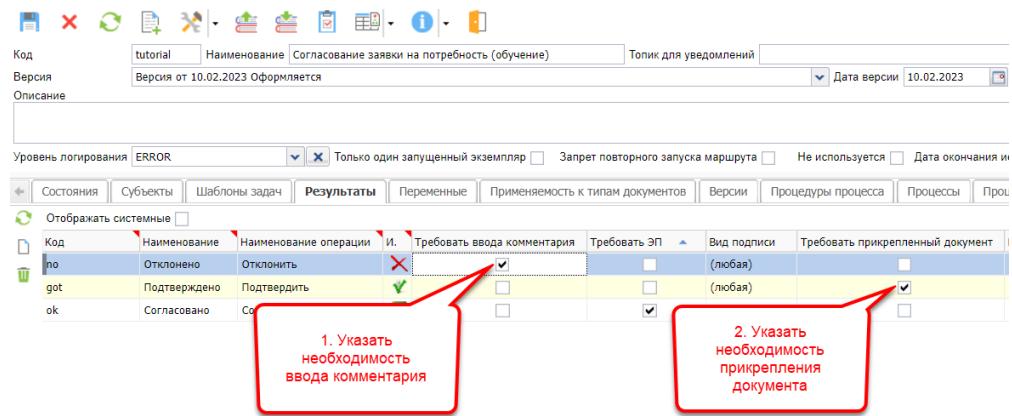


7.4 Задание 4. Дополнительная настройка шаблонов задач, субъектов и результатов процесса

Настройте вставку наименования согласовываемого документа в текст задачи. Настройте требование ввода комментария при отклонении задачи и прикрепления файла при согласовании. Настройте возможность ручного выбора сотрудников, которые получат задачу по маршруту.

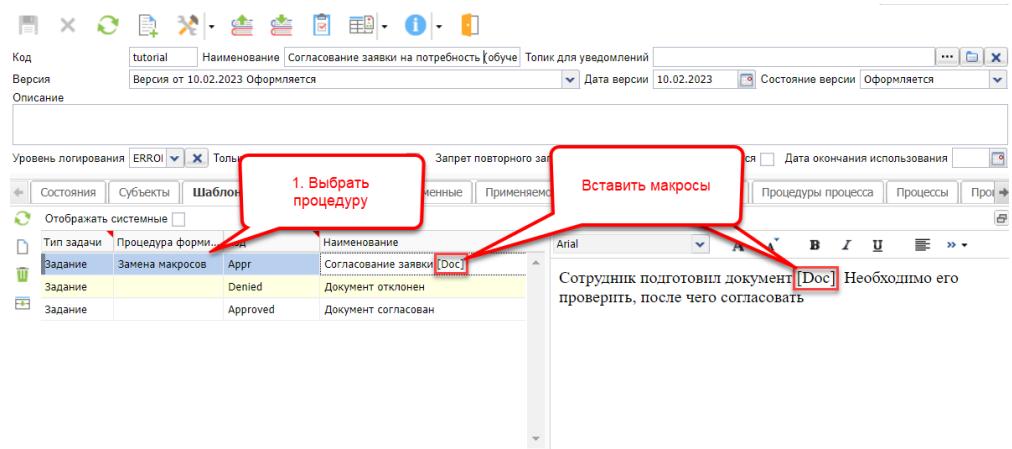
Для настройки требования ввода комментария или прикрепления файла необходимо:

- Перейти на закладку «Результаты» (либо открыв карточку результата от потока графической схемы аналогично пункту 3 выше) и для требуемого результата отметить соответствующие признаки. При выполнении задач с данными результатами будет отображено требование ввода комментария или прикрепления документа соответственно.



Для вставки наименования согласовываемого документа в текст задачи необходимо:

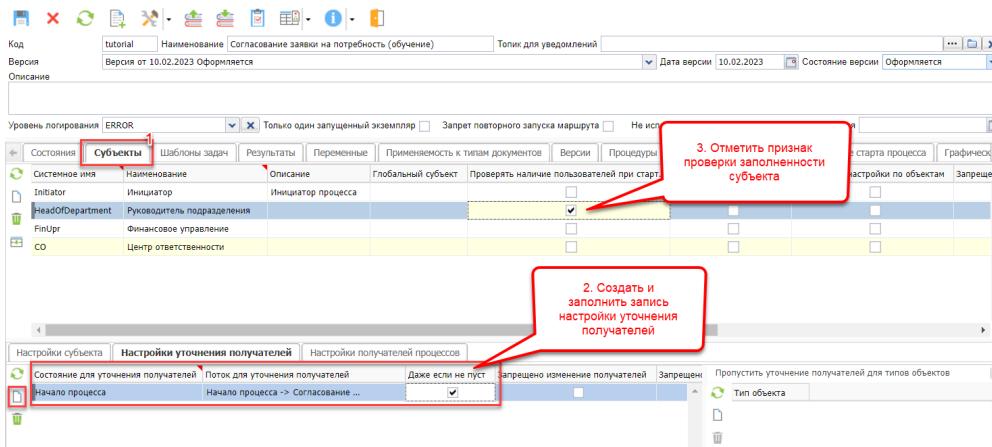
- На закладке «Шаблоны задач» (либо открыв карточку шаблона с графической схемы) указать для шаблона процедуру «Замена макросов» и вставить макросы в текст наименования или текста задачи. При формировании задач по данному шаблону макросы будут заменены на соответствующий текст.



Для настройки возможности ручного выбора сотрудников, которые получат задачу по маршруту, необходимо:

- Выделить требуемого субъекта на закладке «Субъекты» (или на графической схеме). На закладке субъекта «Настройки уточнения получателей» выбрать состояние и поток, при переходе по которому будет производиться уточнение. Отметить признак «Даже если не пуст». Отметить

для субъекта признак «Проверять наличие пользователей при старте процесса». При активации данного потока будет отображено окно уточнения получателей задачи.

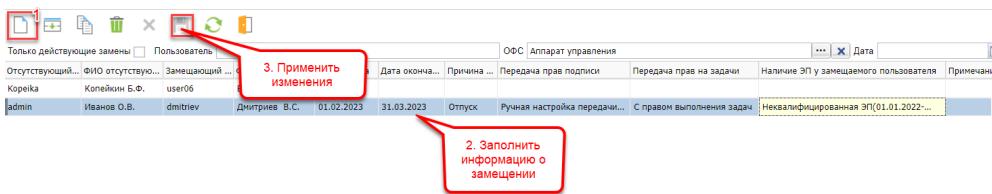


7.5 Задание 5. Работа с замещениями и перенаправлением задач

Настройте замещение. Перенаправьте задачу. Создайте подзадачи.

Для настройки замещения необходимо:

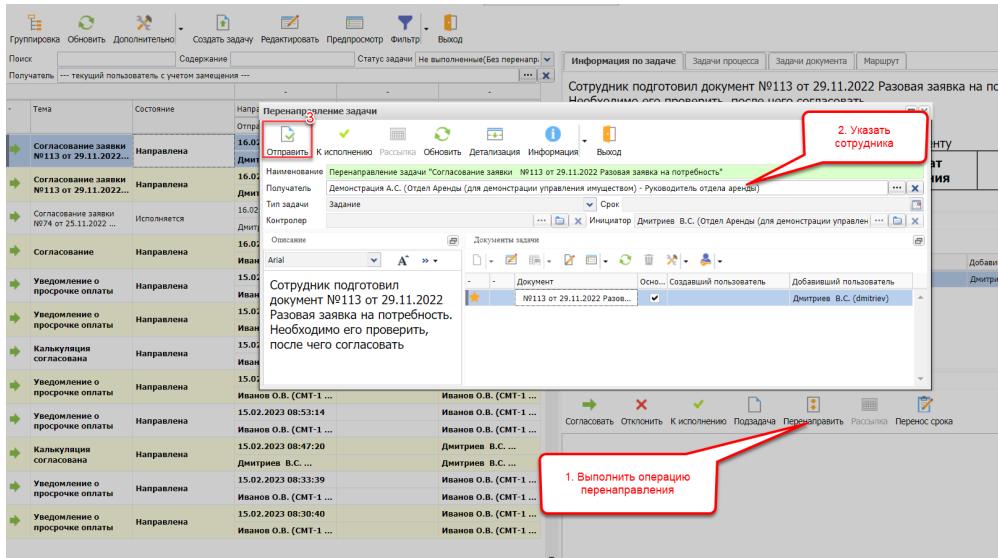
- В приложении «Администратор» открыть перечень замещений (Доступ -> Замещение прав) и выполнить операцию создания. Указать заменяемого и заменяющего сотрудника, применить изменения.



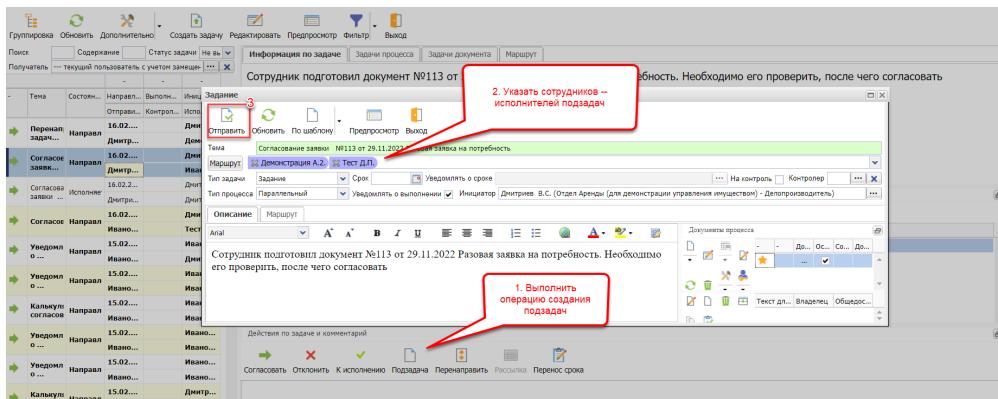
- На закладках «Прикрепленные файлы» и «Документы СЭД» выбрать документы, связанные с замещением (например, скан приказа или соответствующий документ системы)
- На закладке «ЭП без центра сертификации» создать для замещающего пользователя электронную подпись в случае, если ранее у него подписи не было.
- На закладке «Права подписи пользователей» отметить, на какие классы или типы объектов замещающему пользователю будут переданы права (в случае, если ранее в поле «Передача прав подписи» был выбран вариант «Полная передача прав подписи», доступ будет выдан ко всем). На период действия замещения замещающий пользователь в списке своих входящих задач будет видеть и задачи замещающего пользователя (в том числе и направленные до начала действия замещения) и сможет их выполнять в соответствии с настройками замещения. В маршрутах и историях согласования документа будет отображено, что задача была выполнена по праву замещения.

Для подключения к выполнению задачи других сотрудников доступны два инструмента: перенаправление и создание подзадач.

- Для перенаправления задачи необходимо выполнить соответствующую операцию и указать получателя перенаправленной задачи.



- Для создания подзадач необходимо выполнить соответствующую операцию и выбрать сотрудников, которым будут направлены подзадачи. Доступно указание срока исполнения, ввод собственного текста задачи и выбор типа процесса, определяющего, последовательно ли пользователи будут получать задачи или параллельно.

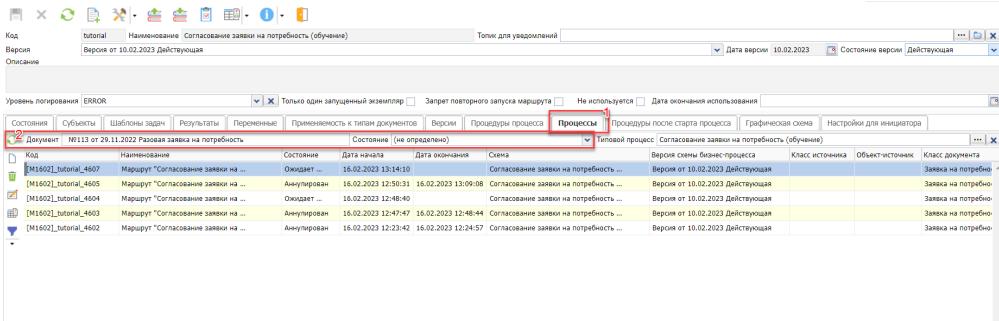


7.6 Задание 6. Анализ запущенных процессов

Просмотрите список процессов по своему маршруту. Узнайте значения переменных маршрута в данный момент. Аннулируйте процесс

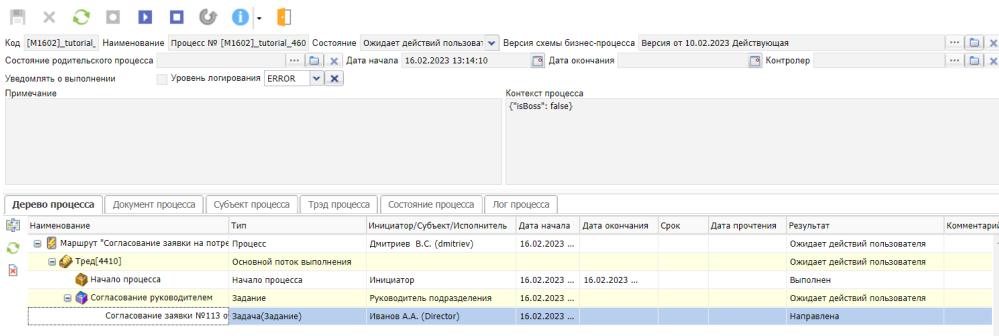
Для выполнения задания необходимо:

- В карточке схемы процесса перейти на закладку «Процессы» и отфильтровать их по документу или состоянию. Перейти в карточку одного из них.



Раздел «Документ процесса» содержит перечень всех переменных процесса и их значений на данный момент.

- На закладке «Документ процесса» расположены документы, от которых запущен процесс, и все позже прикрепленные документы.
- На закладке «Субъект процесса» расположены субъекты процесса и пользователи для каждого из них.
- На закладке «Состояние процесса» приведена история изменений состояний процесса.
- На закладке «Лог процесса» приведен лог процесса, формируемый в зависимости от выбранного уровня логирования.



8 Самостоятельные задания

8.1 Задание 1

Создайте новую схему процесса электронного согласования документа «Выдача книг» (LbrNN_OutOrder), состоящую из трех этапов:

- Согласование Читателем (настройте автоматическое определение сотрудника как руководителя ОФС «Финансовое управление»);
- Обработка Библиотекарем (определите сотрудника на основании атрибута документа);
- Уведомление Инициатора о том, завершился маршрут согласованием или отклонением.

Для каждого этапа предусмотрите возможность отклонения документа, а для обработки Библиотекарем — возможность возврата на корректировку (маршрут не завершается; инициатор исправляет ошибки, после чего задача приходит сразу Библиотекарю без повторного согласования Читателем).

Настройте автоматическую смену состояний в соответствии с прогрессом согласования документа.

8.2 Задание 2

Создайте новую версию созданной в ходе первого урока схемы согласования документа «Выдача книг» (LbrNN_OutOrder), добавив в нее:

- Уточнение субъекта «Библиотекарь»;
- Подписание различными ЭП на каждом этапе согласования;
- Требование прикрепления файла в случае возврата документа на корректировку Библиотекарем.

Настройте замещение с Читателя на другого пользователя.

Запустите процесс и создайте подзадачи от лица Библиотекаря.

9 Практические советы

9.1 Входные параметры процедур

Процедуры процесса

Входными параметрами процедур процесса являются идентификатор процесса `idpProcess` и текстовый параметр `param`, который задается в интерфейсе задания процедуры.

Процедуры состояния

Входными параметрами процедуры состояния являются идентификатор состояния процесса `idpPrState` и текстовый параметр `param`, который задается в интерфейсе задания процедуры.

Процедуры потока

Входными параметрами процедуры потока являются идентификатор состояния процесса `idpPrState`, из которого выходит поток, и идентификатор самого потока `idpPSELFlow`.

Процедуры задач

Процедура формирования текста задачи

Входным параметром процедуры формирования текста задачи является идентификатор задачи `idpTask`.

Процедура, выполняемая после выполнения задачи

Входными параметрами процедуры, выполняемой после выполнения задачи, являются идентификатор задачи `idpTask` и текстовый параметр `param`, который задается в интерфейсе задания процедуры.

9.2 Зарезервированные слова в jexl-скриптах процедур

Кроме служебных слов, которые можно посмотреть в документации к jexl (<https://commons.apache.org/proper/commons-jexl/reference/syntax.html>), зарезервированными словами в скриптах процедур являются только их входные параметры, которые перечислены выше в разделе Входные параметры.

9.3 Список тэгов, доступных в шаблонах задачи

Список тэгов, которые можно использовать в шаблонах задачи, для подстановки в текст и заголовок задачи нужных данных при ее создании.

- [Process] - наименование процесса текущей задачи,
- [State] - наименование состояния процесса текущей задачи,
- [Subject] - наименование субъекта процесса текущей задачи,
- [Now] - текущие дата и время в формате dd.MM.yyyy HH:mm:ss,
- [Date] - текущая дата в формате dd.MM.yyyy,
- [ParentTask] - наименование исходной задачи,
- [ThreadStartTask] - наименование стартовой задачи потока,
- [SourceComment] - комментарий исходной задачи,
- [ThreadStartComment] - комментарий стартовой задачи потока,
- [CorrectionThreadComments] - комментарии корректировки задачи,
- [Task] - наименование текущей задачи,
- [TaskMC]- мнемокод текущей задачи,
- [TaskURL] - url-ссылка на задачу
- [Doc] - наименование основного документа, прикрепленного к текущей задаче,
- [DocMC]- мнемокод основного документа, прикрепленного к текущей задаче,
- [Doc#attr_name]- значение атрибута attr_name основного документа, прикрепленного к текущей задаче (если attr_name - ссылочный атрибут, то будет получено наименование объекта, на которой ссылается атрибут)

9.4 Подходы для получения данных для обработки внутри процедуры

Общий пример процедуры состояния процесса, который будет использоваться дальше:

```
//метод получения идентификатора процесса idvProc по идентификатору состояния idpPrState
var idvProc = Bpm_ProcLibraryPkg.getProcByState(idpPrState);
//метод загрузки объекта процесса ropProc по его идентификатору idvProc
var ropProc = Bpm_ProcessApi.load(idvProc);
//метод превращения роли объекта в JexlRop для более удобного обращения к атрибутам
//объекта
var jRopProc = new("ru.bitec.app.gtk.jexl.session.JexlRop", ropProc);
//метод получения гида документа процесса gidDoc по идентификатору состояния idpPrState
var gidDoc = Bpm_PrDocApi.getProcDocByPrState(idpPrState);
```

(продолжается на следующей странице)

(продолжение с предыдущей страницы)

```
//загрузка роли документа по его гиду (например, в данном случае это объект класса Wf_Doc)
var ropDoc = Wf_DocApi.loadByGid(gidDoc);
var jRopDoc = new("ru.bitec.app.gtk.jexl.session.JexlRop", ropDoc);
//получим из документа указанное в нем подразделение idDepartment
var idDepartment = jRopDoc.idDepartment;
//пусть в качестве параметра в процедуру мы передали системное имя нужного нам подразделения 'Отдел1', присвоим его в переменную
var svOurDepartment = param;
//получим идентификатор нашего подразделения с помощью метода нахождения его по мнемокоду
var idvOurDepartment = Bs_DepartmentApi.findbyMnemoCode(svOurDepartment);
//сравним подразделение в документе и наше, и положим в булеву переменную
var isOurDepartment = idDepartment == idvOurDepartment;
//системное имя переменной
var spVarCode = "bNeedHeadPersonSign";
//положим полученное значение в переменную процесса
Bpm_ProcessApi.setProcessVar(ropProc, spVarCode, isOurDepartment);
//если нужное нам подразделение, то заполним начальником подразделения соответствующий субъект
if (isOurDepartment) {
    //системное имя субъекта процесса
    var svSubj = "HeadPerson";
    //получение субъекта по его системному имени
    var ropTargetOpt = Bpm_PrSubjectApi.getByPSSubject(ropProc, svSubj);
    //проверка наличия нашего субъекта в схеме
    if (ropTargetOpt.isDefined()) {
        //получаем субъект, если он есть
        var ropTarget = ropTargetOpt.get();
        //запишем SQL-запрос к БД для получения идентификатора пользователя начальника нашего подразделения
        var query = `select coalesce (t.id,0) as iduser
                    from btk_user t
                    join bs_person t1 on t.id = t1.iduser
                    join bs_department t2 on t2.idmngperson = t1.id
                    where t2.id = ` + toString(idDepartment);
        //выполним запрос, и результат представим в виде коллекции значений
        var l = sql(query).asList();
        //обойдем получившуюся коллекцию значений
        for (w:l){
            //обращаемся к атрибуту, который мы получили из запроса, следующим образом
            //iduser
            //кидаем ошибку, если пользователя не нашли
            if (w.iduser == 0) {
                raise("Не найден пользователь");
            }
            //если нашли, регистрируем пользователя в субъект
            Bpm_PrSubjUserApi.register(ropTarget, w.iduser, OB, OB);
        }
    } else {
        raise("Субъект " + svSubj + " не найден. Обратитесь к администратору");
    }
}
```

В данной процедуре происходит сравнение атрибута Подразделение в документе процесса (документ, в данном случае, является объектом класса Wf_Doc) с переданным, в качестве параметра в процедуру, подразделением. Результат их сравнения записывается в переменную процесса. Если подразделения совпадают, то с помощью SQL-запроса из базы данных берется пользователь, являющийся начальником данного подразделения, и этот пользователь регистрируется в субъект процесса.

Некоторые обязательные действия, которые всегда должны присутствовать в процедурах

Практически в каждой процедуре необходимо использовать методы получения процесса и методы получения документа процесса.

Методы получения процесса

```
//метод получения идентификатора процесса idvProc по идентификатору состояния idpPrState
//для процедур состояния и потока
var idvProc = Bpm_ProcLibraryPkg.getProcByState(idpPrState);
//метод получения идентификатора процесса idvProc по идентификатору задачи idpTask (для
//процедур задач)
var idvProce = Bpm_TaskApi.getProcessId(idpPrState);
```

Получение данных текущего запущенного процесса

Методы получения идентификатора процесса описаны выше в разделе Методы получения процесса. При наличии идентификатора процесса можно загрузить его объект с помощью вызова метода load из Api класса:

```
//метод загрузки объекта процесса ropProc по его идентификатору idvProc
var ropProc = Bpm_ProcessApi.load(idvProc);
```

Для более удобного обращения к атрибутам загруженного объекта необходимо преобразовать ропу объекта в JexlRop следующим образом:

```
//метод превращения ропы объекта в JexlRop для более удобного обращения к атрибутам
//объекта
var jRopProc = new("ru.bitec.app.gtk.jexl.session.JexlRop", ropProc);
```

Обращение к атрибутам такого объекта производится следующим образом: jRopProc.имя_атрибута.

Получение данных внутри процедур из разных объектов

Загрузка объектов в скриптах может производиться следующим образом (на примере сверху):

```
//метод загрузки объекта по идентификатору с помощью метода load
var ropProc = Bpm_ProcessApi.load(idvProc);

//метод загрузки объекта по его гиду с помощью метода loadByGid (например, в данном
//случае это объект класса Wf_Doc)
var ropDoc = Wf_DocApi.loadByGid(gidDoc);
```

Для нахождения идентификатора объекта по его мнемокоду используется метод findByMnemoCode из Api класса объекта:

```
//получим идентификатор нашего подразделения с помощью метода нахождения его по мнемокоду
var idvOurDepartment = Bs_DepartmentApi.findbyMnemoCode(svOurDepartment);
```

Также получить необходимые данные можно с помощью sql-запросов:

```
//запишем SQL-запрос к БД для получения идентификатора пользователя начальника нашего подразделения
var query = `select coalesce (t.id,0) as iduser
from btk_user t
join bs_person t1 on t.id = t1.iduser
join bs_department t2 on t2.idmngperson = t1.id
where t2.id = ` + toString(idDepartment);
//выполним запрос, и результатом представим в виде коллекции значений
var l = sql(query).asList();
```

Все возможные методы выполнения sql-запросов можно посмотреть в контекстной помощи jexl-скрипта. Найти ее можно по пути: пункт меню Сервисы - Инструменты - Выполнить jexl-скрипт, в открывшемся окне выполнить операцию Помощь.

Методы получения документа процесса и работы с его атрибутами и коллекциями

```
//метод получения гида документа процесса gidDoc по идентификатору состояния idpPrState
var gidDoc = Bpm_PrDocApi.getProcDocByPrState(idpPrState);

//загрузим объект по его гиду с помощью метода loadByGid (например, в данном случае это объект класса Wf_Doc)
var ropDoc = Wf_DocApi.loadByGid(gidDoc);
//метод превращения роли объекта в JexlRop для более удобного обращения к атрибутам объекта
var jRopProc = new("ru.bitec.app.gtk.jexl.session.JexlRop", ropDoc);
//получим идентификаторы объектов версии документа Wf_DocVer, данный класс является коллекцией к Wf_Doc
var listDocVer = sql(`select t.id as idverdoc
from wf_docver t
where t.idDoc = ` + toString(ropDoc.id));
//обойдем все полученные идентификаторы
for (i:listDocVer) {
    //загрузим объекты по идентификатору
    var ropDocVer = Wf_DocVerApi.load(i.idverdoc)
    var jRopDocVer = new("ru.bitec.app.gtk.jexl.session.JexlRop", ropDocVer);
    //изменим атрибут "Описание" в полученном объекте, например, добавим строчку "Новая строка в описании"
    var svDesc = jRopDocVer.sDescription + " Новая строка в описании";
    Wf_DocVerApi.setsDescription(ropDocVer, svDesc);
    //для изменения ссылочных атрибутов объекта необходимо указывать именно идентификаторы объекта, на который ссылается атрибут, т. е. id или gid
    //например, при задании "Состояния" объекта idState необходимо указывать соответствующий id в классе Btk_ClassState, а при задании "Формата бумаги" gidFormatPaper - gid соответствующего объекта из класса Btk_Object
}
```

Работа с переменными процесса

Метод получения значения переменной процесса

```
//метод получения значения переменной процесса value по ропу процесса ropProc и  
//системному имени переменной процесса spVarCode  
var value = Bpm_ProcessApi.getProcessVar(ropProc, spVarCode);
```

Метод установки значения переменной процесса

```
//метод установки значения переменной процесса value по ропу процесса ropProc и  
//системному имени переменной процесса spVarCode  
Bpm_ProcessApi.setProcessVar(ropProc, spVarCode, value);  
  
//из примера выше  
//сравним подразделение в документе и наше, и положим в булеву переменную  
var isOurDepartment = idDepartment == idvOurDepartment;  
//системное имя переменной  
var spVarCode = "bNeedHeadPersonSign";  
//положим полученное значение в переменную процесса  
Bpm_ProcessApi.setProcessVar(ropProc, spVarCode, isOurDepartment);
```

Получение текущей версии схемы бизнес-процесса

```
//метод получения текущей версии схемы БП, который принимает на вход роп процесса, и  
//возвращает роп версии схемы БП  
var ropPSVersion = Bpm_PSVersionApi.getByProcess(ropProc);
```

Получение задач процесса

Получить весь список задач процесса можно с помощью метода

```
`Bpm_TaskApi.getTaskByProcess(idvProc)` ,
```

в который необходимо передать идентификатор процесса. Из метода вернется список ропов задачи.

Пример:

```
//получим все задачи процесса  
var list = asJava(Bpm_TaskApi.getTaskByProcess(idvProc));  
//обойдем весь список задач  
for (i:list) {  
    //преобразуем вернувшиеся ропы в JexlRop  
    var jRopTask = new("ru.bitec.app.gtk.jexl.session.JexlRop", i);  
    //...  
  
    //для нахождения выполненных задач можно сравнить состояние задачи с состоянием  
    //Выполнено  
    //получим идентификатор состояния "Выполнено" для задач
```

(продолжается на следующей странице)

(продолжение с предыдущей страницы)

```
var idvStateDone = Btk_ClassStateApi.findByNameAndIdClass("Bpm_StateDone", Btk_
ClassApi.findByIdMnemoCode("Bpm_Task"), OB);
if (jRopTask.idState == idvStateDone) {
    //если состояние данной задачи "Выполнено"
    /**
}
}
```

Задание условия пропуска этапа

Для этапов с типом «Необязательное задание» в схеме маршрута необходимо указывать условия пропуска. Условие пропуска этапа должно содержать в себе jexl-код, возвращающий булево значение. Если jexl-код вернул true - то данный этап пропускается, если false - то выполняется. Для того, чтобы условие было максимально простым, в нем можно обращаться к переменным процесса.

```
bNeedHeadPersonSign == false //в этом случае этап будет пропущен, если переменная
→процесса bNeedHeadPersonSign равна false
```

Динамическое назначение исполнителей по задаче для определенного этапа

Для задания исполнителей по задаче определенного этапа необходимо зарегистрировать пользователей в соответствующий субъект процесса с помощью метода

```
`Bpm_PrSubjUserApi.register` ,
```

который принимает два параметра: гор субъекта процесса и идентификатор пользователя.

```
//системное имя субъекта процесса
var svSubj = "HeadPerson";
//получение субъекта по его системному имени
var ropTargetOpt = Bpm_PrSubjectApi.getByPSSubject(ropProc, svSubj);
//проверка наличия нашего субъекта в схеме
if (ropTargetOpt.isDefined()) {
    //получаем субъект, если он есть
    var ropTarget = ropTargetOpt.get();
    //запишем SQL-запрос к БД для получения идентификатора начальника
→нашего подразделения
    var query = `select coalesce (t.id,0) as iduser
        from btk_user t
        join bs_person t1 on t.id = t1.iduser
        join bs_department t2 on t2.idmngperson = t1.id
        where t2.id = ` + toString(idDepartment);
    //выполним запрос, и результатом представим в виде коллекции значений
    var l = sql(query).asList();
    //обойдем получившуюся коллекцию значений
    for (w:l){
        //обращаемся к атрибуту, который мы получили из запроса, следующим образом w.
→iduser
        //кидаем ошибку, если пользователя не нашли
        if (w.iduser == 0) {raise("Не найден пользователь");}
    }
}
```

(продолжается на следующей странице)

(продолжение с предыдущей страницы)

```

    //если нашли, регистрируем пользователя в субъекте
    Bpm_PrSubjUserApi.register(ropTarget, w.iduser);
}
} else {
    raise("Субъект " + svSubj + " не найден. Обратитесь к администратору");
}

```

9.5 Случаи, когда в субъекте несколько пользователей

- Если для одного субъекта процесса задано несколько исполнителей, тогда для каждого из исполнителей будет сгенерирована задача.

The screenshot shows a task list with the following details:

| Наименование | Инициатор/Субъект/Исполнитель | Дата начала | Дата окончания |
|--|---|-------------|---------------------|
| Маршрут: «Согласование исходящего письма» | Адрианов А.А. (AdrianaAA) | 02.11.2023 | |
| Согласование начальником сектора | Начальник сектора | 02.11.2023 | 02.11.2023 13:08:53 |
| Требуется согласовать письмо Тест начальнику сектора | Адрианов А.А. (admin) за Лугуменев А.С. (LugumenevAS) | 02.11.2023 | 02.11.2023 13:08:53 |
| Согласование начальником отдела | Начальник отдела | 02.11.2023 | 02.11.2023 13:09:05 |
| Требуется согласовать письмо Тест начальнику отдела | Адрианов А.А. (admin) за Свиридов А.В. (SviridovAV) | 02.11.2023 | 02.11.2023 13:09:04 |
| Согласование согласующими лицами: 4ый ур. согласования | Согласующие лица 4 | 02.11.2023 | |
| Требуется согласовать письмо Тест согласующим лицам | Свиридов А.В. (SviridovAV) | 02.11.2023 | |
| Требуется согласовать письмо Тест согласующим лицам | Прилуцкий П.Н. (PrlutskiyPN) | 02.11.2023 | |
| Требуется согласовать письмо Тест согласующим лицам | Гаврилов С.В. (GavrilovSV) | 02.11.2023 | |
| Требуется согласовать письмо Тест согласующим лицам | Каплин И.В. (KaplinIV) | 02.11.2023 | |
| Требуется согласовать письмо Тест согласующим лицам | Никищенко Ю.И. (NikishenkoYI) | 02.11.2023 | |
| Не запущено Согласование согласующими лицами: 3ий ур. согласования | Согласующие лица 3 | | |
| Не запущено Согласование согласующими лицами: 2ий ур. согласования | Согласующие лица 2 | | |
| Не запущено Согласование согласующими лицами: 1ый ур. согласования | Согласующие лица 1 | | |
| Не запущено Согласование подписанта | Подписанты | | |
| Прилуцкий П.Н. | | | |
| Не запущено Отправка письма | Отправитель | | |
| Герус С.Г. | | | |

Как видно на рисунке выше, для состояния «Согласование согласующими лицами: 4ый ур. согласования» задано пять согласующих лиц, каждый из согласующих получает личную задачу на согласование документа.

- Задачи всегда направляются параллельно, но есть возможность настроить состояния маршрута таким образом, чтобы исполнители получали задачи последовательно. Для примера разберем маршрут «Согласование распоряжения».

Для того, чтобы определить, будет согласование происходить параллельно или последовательно, для документа была добавлена и выведена в фрейм характеристика объектная характеристика «Параллельное согласование».

The screenshot shows the 'Реквизиты распоряжения' (Properties of the Order) form with the 'Характеристики' (Characteristics) tab selected. In the 'Характеристики' section, there is a checkbox labeled 'Параллельное согласование' (Parallel Approval), which is checked.

По умолчанию значение характеристики = 1, что предполагает запуск маршрута с параллельным согласованием. Пользователь запускающий маршрут сам определяет, запустить маршрут с последовательным или параллельным согласованием, изменения значение данной характеристики.

В состоянии маршрута «Определение начальных данных» при помощи процедуры «Получение значения атрибута в переменную» происходит определение переменной «Маршрут параллельный» (isParallel), а также происходит заполнение субъекта согласующих лиц (Coordinator) при помощи процедуры «Обновление состава согласующих лиц из документа».

| Состояния | Субъекты | Шаблоны задач | Результаты | Переменные | Применимость к типам документов | Версия | Процедуры процесса | Процессы | П | | | | | | | |
|--|--|-----------------------------------|---------------------------|------------|---------------------------------|--------|--------------------|-----------------------------------|---|--|--|--|--|--|--|--|
| Тип состояния БП | Порядковый номер | Наименование | Условие пропуска | | | | | Отображать | | | | | | | | |
| Аннулирование процесса | -1 | Аннулирование процесса | | | | | | | | | | | | | | |
| Начало процесса | 1 | Начало процесса | | | | | | | | | | | | | | |
| Сценарий | 2 | Определение начальных данных | | | | | | | | | | | | | | |
| Возврат при исправлении замечаний | 5 | Исправление замечаний | | | | | | | | | | | | | | |
| Задание | 6 | Исправление замечаний согласующих | | | | | | | | | | | | | | |
| Эксклюзивный шлюз | 10 | Начальник? | | | | | | | | | | | | | | |
| Необязательное задание | 20 | Согласование начальником | isEmptyHeadPerson == true | | | | | | | | | | | | | |
| Эксклюзивный шлюз | 21 | Согласование параллельно? | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | | |
| Сходящие потоки Входящие потоки Процедуры ЖЦ документа на состояние Области подписания Используемые типы связи | | | | | | | | | | | | | | | | |
| процедуры до входа в состояние | | | | | | | | | | | | | | | | |
| Порядок выполне... | Процедура | Параметр | | | | | | Процедуры после входа в состояние | | | | | | | | |
| 60 | Обновление подписчиков из документов процесса | Sign | | | | | | | | | | | | | | |
| 70 | Проверка всех подписчиков на участие в электронном ... | isSignerElectronic | | | | | | | | | | | | | | |
| 80 | Проверка отсутствия пользователей в субъекте процесса. ... | Coordinator;isEmptyCo | | | | | | | | | | | | | | |
| 90 | Удаление из субъекта пользователей присутствующих в других ... | HeadPerson;Sign | | | | | | | | | | | | | | |
| 100 | Проверка отсутствия пользователей в субъекте процесса. ... | HeadPerson;isEmptyHe | | | | | | | | | | | | | | |
| 110 | Получение значения атрибута в переменную | bParallel;isParallel | | | | | | | | | | | | | | |

Далее в состоянии «Согласование параллельно?» с типом эксклюзивный шлюз, в зависимости от значения переменной, определяется последовательно или параллельно документ будут согласовывать согласующие лица.

| Состояния | Субъекты | Шаблоны задач | Результаты | Переменные | Применимость к типам документов | Версия | Процедуры процесса | Процессы | П |
|--|---------------------------|-----------------------------------|----------------------------|-------------------|---------------------------------|--------|--------------------|-------------|---|
| Тип состояния БП | Порядковый номер | Наименование | Условие пропуска | | | | | Отобра | |
| Задание | 6 | Исправление замечаний согласующих | | | | | | | |
| Эксклюзивный шлюз | 10 | Начальник? | | | | | | | |
| Необязательное задание | 20 | Согласование начальником | isEmptyHeadPerson == true | | | | | | |
| Эксклюзивный шлюз | 21 | Согласование параллельно? | | | | | | | |
| Эксклюзивный шлюз | 23 | Проверка пользователей | | | | | | | |
| Задание | 24 | Последовательное согласование | | | | | | | |
| Необязательное задание | 30 | Согласование согласующими лицами | isEmptyCoordinator == true | | | | | | |
| Эксклюзивный шлюз | 35 | Подписант работает в системе? | | | | | | | |
| | | | | | | | | | |
| Сходящие потоки Входящие потоки Процедуры ЖЦ документа на состояние Области подписания Используемые типы связи | | | | | | | | | |
| Конечное состояние | Процедура на вход в поток | Порядковы... | Условие входа в поток | | | | | Состояние д | |
| Проверка пользователей | | | 1 | isParallel != "1" | | | | | |
| Согласование согласующими лицами | | | 2 | isParallel == "1" | | | | | |

Если значение переменной не равно 1, маршрут переходит в поток «Проверка пользователей», в котором будет осуществлено последовательное согласование документа.

На данном этапе в субъекте «Согласующие» (Coordinator) может находиться несколько исполнителей, и при направлении задачи по данному субъекту, задачи направляются всем исполнителям параллельно. Для этого реализовано следующее состояние «Проверка пользователя», в котором циклически будет отбираться один исполнитель из субъекта «Согласующие», и только для него будет направляться задача. Цикл будет повторяться, пока задачи не будут направлены всем исполнителям.

В состоянии «Проверка пользователей» последовательно выполняются две процедуры:

- Процедура перемещения первого пользователя субъекта в другой субъект – данная процедура берет первого исполнителя из субъекта «Согласующие» (Coordinator) и перемещает его в другой субъект «Последовательные согласующие» (NumCoordinator). В дальнейшем задача будет направлена для субъекта «Последовательные согласующие», в котором будет находиться только один исполнитель.

- Проверка отсутствия пользователей в субъекте процесса. Результат записывается в переданную переменную – данная процедура производит проверку на наличие исполнителей в субъекте «Последовательные согласующие» (NumCoordinator), и записывает значение (1 – отсутствует или 0 – пользователь задан в субъекте) в переменную «Отсутствует последовательный согласующий» (isEmptyNumCoordinator).

| Составления | | | | | | |
|---------------------------------|------------------|---|----------------------------|---------------------------------------|-------------------------------------|------------------------------|
| Субъекты | | Шаблоны задач | | Результаты | | Перенесенные |
| Применимость к типам документов | | Версии | | Процедуры процесса | | Процессы |
| Графическая схема | | Процедуры после старта процесса | | Процедуры после старта процесса | | |
| Тип состояния БП | Порядковый номер | Название | Условие пропуска | Отображать как событие ссылку | Контролируемое | Субъект процесса |
| Задание | 6 | Изправление замечаний согласующих | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Инициатор |
| Экспозиционный шаблон | 10 | Начальник? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Начальник |
| Неизбательное задание | 20 | Согласование начальником | isEmptyHeadPerson == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Экспозиционный шаблон | 21 | Согласование параллельно? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Экспозиционный шаблон | 23 | Проверка пользователя | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Задание | 24 | Последовательное согласование | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Последовательные согласующие |
| Неизбательное задание | 30 | Согласование согласующими лицами | isEmptyCoordinator == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Согласующие |
| Экспозиционный шаблон | 35 | Подписант работает в системе? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| | | | | | | |
| Создание потоки | | Входящие потоки | | Процедуры | | ЖЦ документа на состоянии |
| Области подписания | | Использованные типы связи | | Процедуры после входа в состояние | | |
| П. | | Процедура | | Параметр | | |
| 10 | | Процедура перенесения первого пользователя субъекта в ... | | Coordinator:NumCoordinator | | |
| 20 | | Проверка отсутствия пользователя в субъекте процесса... | | NumCoordinator:isEmpty/NumCoordinator | | |
| | | | | | | |

Если значение переменной «Отсутствует последовательный согласующий» = 0 (значит в субъекте есть пользователь и на него можно направить задачу), происходит переход в состояние «Последовательное согласование», на котором исполнителю, перемещенному в субъект «Последовательные согласующие» (NumCoordinator), будет направлена задача.

| Составления | | | | | | |
|---------------------------------|------------------|-----------------------------------|----------------------------|-------------------------------------|-------------------------------------|------------------------------|
| Субъекты | | Шаблоны задач | | Результаты | | Перенесенные |
| Применимость к типам документов | | Версии | | Процедуры процесса | | Процессы |
| Графическая схема | | Процедуры после старта процесса | | Процедуры после старта процесса | | |
| Тип состояния БП | Порядковый номер | Название | Условие пропуска | Отображать как событие ссылку | Контролируемое | Субъект процесса |
| Задание | 6 | Изправление замечаний согласующих | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Инициатор |
| Экспозиционный шаблон | 10 | Начальник? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Начальник |
| Неизбательное задание | 20 | Согласование начальником | isEmptyHeadPerson == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Экспозиционный шаблон | 21 | Согласование параллельно? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Экспозиционный шаблон | 23 | Проверка пользователя | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Задание | 24 | Последовательное согласование | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Последовательные согласующие |
| Неизбательное задание | 30 | Согласование согласующими лицами | isEmptyCoordinator == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Согласующие |
| Экспозиционный шаблон | 35 | Подписант работает в системе? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| | | | | | | |
| Создание потоки | | Входящие потоки | | Процедуры | | ЖЦ документа на состоянии |
| Области подписания | | Использованные типы связи | | Процедуры после выхода из состояния | | |
| Конечное состояние | | Процедура на вход в поток | | Порядковый | | |
| Последовательное согласование | | Условие входа в поток | | Состояние для исправлений | | Основной сценарий |
| Подписант работает в системе? | | 1 isEmptyNumCoordinator == false | | | | |
| | | 2 isEmptyNumCoordinator == true | | | | |

Если значение переменной «Отсутствует последовательный согласующий» = 1 (для субъекта не указан пользователь, что предполагает, что все исполнители завершили последовательное согласование), происходит переход в состояние «Подписант работает в системе?», что по сути является выходом из последовательного согласования на маршруте.

На состоянии «Последовательное согласование» согласующему из субъекта «Последовательные согласующие» приходит задача на согласование. В случае успешного согласования, выполняется процедура «Процедура перемещения первого пользователя субъекта в другой субъект», которая перемещает исполнителя из субъекта «Последовательные согласующие» (NumCoordinator) в субъект «Архив согласующих» (CoordinatorArchive).

| Составления | | | | | | |
|---------------------------------|------------------|---------------------------------------|----------------------------|-------------------------------------|-------------------------------------|------------------------------|
| Субъекты | | Шаблоны задач | | Результаты | | Перенесенные |
| Применимость к типам документов | | Версии | | Процедуры процесса | | Процессы |
| Графическая схема | | Процедуры после старта процесса | | Процедуры после старта процесса | | |
| Тип состояния БП | Порядковый номер | Название | Условие пропуска | Отображать как событие ссылку | Контролируемое | Субъект процесса |
| Неизбательное задание | 20 | Согласование начальником | isEmptyHeadPerson == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Начальник |
| Экспозиционный шаблон | 21 | Согласование параллельно? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Экспозиционный шаблон | 23 | Проверка пользователя | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Задание | 24 | Последовательное согласование | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Последовательные согласующие |
| Неизбательное задание | 30 | Согласование согласующими лицами | isEmptyCoordinator == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Согласующие |
| Экспозиционный шаблон | 35 | Подписант работает в системе? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | |
| Задание | 36 | Согласование подписантами вне системы | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Центр регистрации |
| Задание | 40 | Согласование подписантами в системе | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Подписанты |
| | | | | | | |
| Создание потоки | | Входящие потоки | | Процедуры | | ЖЦ документа на состоянии |
| Области подписания | | Использованные типы связи | | Процедуры после выхода из состояния | | |
| П. | | Процедура | | Порядок выполн... | | |
| 10 | | Процедура перенесения первого ... | | CoordinatorArchive | | |
| | | | | | | |

Исполнители, перемещенные в субъект «Архив согласующих» (CoordinatorArchive), далее не участвуют в согласовании, т.к. свою задачу по согласованию они выполнили. Маршрут при этом возвращается на состояние «Проверка пользователей», на котором повторно происходит перемещение следующего исполнителя из субъекта «Согласующие» (Coordinator) в субъект «Последовательные согласующие» (NumCoordinator). Последовательное согласование закончится тогда, когда в субъекте «Согласующие» (Coordinator) не останется пользователей, после чего процедура «Проверка отсутствия пользователей в субъекте процесса». Результат записывается в переданную переменную вернет значение переменной «Отсутствует последовательный согласующий» = 1, и по условиям эксклюзивного шлюза маршрут выйдет из данного цикла.

| Составление | | | | | | | | |
|---------------------------|------------------|---------------------------------------|----------------------------|-------------------------------|-------------------------------------|------------------------------|---|--|
| Субъекты | | | | | | | | |
| Шаблоны задач | | | | | | | | |
| Тип состояния БП | Порядковый номер | Название | Условие пропуска | Отображать как событие ссылку | Контролируемое | Субъект процесса | О | |
| Необязательное задание | 20 | Согласование начальнику | isEmptyHeadPerson == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Начальник | | |
| Эксклюзивный шлюз | 21 | Согласование параллельно? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | | | |
| Эксклюзивный шлюз | 23 | Проверка пользователей | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | | | |
| Задание | 24 | Последовательное согласование | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Последовательные согласующие | | |
| Необязательное задание | 30 | Согласование согласующими лицами | isEmptyCoordinator == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Согласующие | | |
| Эксклюзивный шлюз | 35 | Подписанкт работает в системе? | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | | | |
| Задание | 36 | Согласование подписанктов вне системы | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Центр регистрации | | |
| Задание | 40 | Согласование подписанктов в системе | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Подписанты | | |
| Исходящие потоки | | | | | | | | |
| Входящие потоки | | | | | | | | |
| Процедуры | | | | | | | | |
| ЭЖ документа на состоянии | | | | | | | | |
| Области подписания | | | | | | | | |
| Использование типов связи | | | | | | | | |
| Конечное состояние | | | | | | | | |
| Процедура на вход в поток | | | | | | | | |
| Порядковый | | | | | | | | |
| Условие входа в поток | | | | | | | | |
| Состояние для истрав... | | | | | | | | |
| Основной сценарий | | | | | | | | |
| Результат выполнения | | | | | | | | |
| Условие | | | | | | | | |
| Хотя бы один | | | | | | | | |
| Основной поток | | | | | | | | |
| Шаблон задачи | | | | | | | | |

- При выполнении параллельных задач, итоговый результат определяется по настроенным на исходящих переходах условиях.

На выбор итогового потока при выполнении задач исполнителями влияют такие параметры потока, как «Порядковый номер» и «Условие».

В условии есть выбор из трех опций:

- Все – чтобы перейти в поток с данным условием, все исполнители должны выполнить задачу с одинаковым результатом, заданным на потоке в поле «Результат выполнения». Выполнение задачи хотя бы одним из исполнителей с отличным результатом приводит к невозможности входа в данный поток;
- Хотя бы один – чтобы перейти в поток с данным условием, хотя бы один из исполнителей должен выполнить задачу с заданным на потоке результатом, при этом задачи остальных исполнителей будут автоматически аннулированы, и маршрут перейдет в соответствующий поток;
- Хотя бы один (ждать всех) – чтобы перейти в поток с данным условием, хотя бы один из пользователей должен выполнить задачу с заданным на потоке результатом, при этом система будет ждать, пока остальные исполнители также выполнят свои задачи, и только после выполнения задач всеми исполнителями произойдет переход в заданный поток. Если для состояния настроено несколько исходящих потоков с условием «Хотя бы один (ждать всех)», после выполнения всех задач исполнителями, система автоматически выберет более приоритетный поток, ориентируясь на «Порядковый номер» потока, который также является приоритетом потоков, где 1 – максимальный приоритет. Условие «Хотя бы один» всегда является условием с максимальным приоритетом, при выполнении задачи с результатом, заданным для потока с условием «Хотя бы один», все остальные задачи аннулируются, и маршрут идет по данному потоку.

| Составление | | | | | | | | |
|---------------------------|------------------|---|-----------------------------|-------------------------------|-------------------------------------|----------------------|---|--|
| Субъекты | | | | | | | | |
| Шаблоны задач | | | | | | | | |
| Тип состояния БП | Порядковый номер | Название | Условие пропуска | Отображать как событие ссылку | Контролируемое | Субъект процесса | О | |
| Задание | 203 | Исправление замечаний инициаторов | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Инициатор | | |
| Задание | 204 | Исправление замечаний инициаторов | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Инициатор | | |
| Необязательное задание | 210 | Согласование согласующими лицами: 1ый ур... | isEmptyCoordinator1 == true | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Согласующие лица1 | | |
| Эксклюзивный шлюз | 215 | АЧР | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | | | |
| Задание | 216 | Исправление замечаний инициаторов | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Инициатор | | |
| Задание | 220 | Исправление замечаний начальником отдела | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Начальник отдела | | |
| Необязательный шлюз | 221 | Если начальник не ответил | | <input type="checkbox"/> | <input checked="" type="checkbox"/> | Максимальная попытка | | |
| Исходящие потоки | | | | | | | | |
| Входящие потоки | | | | | | | | |
| Процедуры | | | | | | | | |
| ЭЖ документа на состоянии | | | | | | | | |
| Области подписания | | | | | | | | |
| Использование типов связи | | | | | | | | |
| Конечное состояние | | | | | | | | |
| Процедура на вход в поток | | | | | | | | |
| Порядковый | | | | | | | | |
| Условие входа в поток | | | | | | | | |
| Состояние для истрав... | | | | | | | | |
| Основной сценарий | | | | | | | | |
| Результат выполнения | | | | | | | | |
| Условие | | | | | | | | |
| Хотя бы один (жд... | | | | | | | | |
| Основной поток | | | | | | | | |
| Лицо (Документ ...) | | | | | | | | |
| Шаблон задачи | | | | | | | | |

Разберем пример настроенного маршрута. Для состояния «Согласование согласующими лицами: 1ый ур. согласования» задано три исходящих потока.

Для потока №1 настроено условие «Хотя бы один (ждать всех)» – данное условие означает, что при выборе результата выполнения задачи «Отклонено» хотя бы одним из исполнителей, маршрут пойдет по данному потоку, но прежде чем войти в данный поток, все остальные исполнители должны выполнить свою задачу.

Для перехода №2 настроено условие «Все» – чтобы маршрут перешел в данный поток, необходимо чтобы все исполнители выполнили задачу с результатом «Согласовано согласующими лицами». Выполнение задачи с результатом отличным от «Согласовано согласующими лицами» хотя бы одним из исполнителем заблокирует возможность перейти в данный поток.

Для перехода №3 настроено условие «Хотя бы один (ждать всех)» – аналогично потоку №1, чтобы маршрут перешел в данный поток, хотя бы один исполнитель должен выполнить задачу с результатом «Согласовано согласующими лицами с замечаниями». Так как на состоянии настроено два потока с условием «Хотя бы один (ждать всех)», приоритет входа в поток определяется по порядковому номеру потока.

Это означает, что если один из пользователей выполнит задачу с результатом «Отклонено», а другой пользователь выполнит задачу с результатом «Согласовано согласующими лицами с замечаниями», то маршрут перейдет в поток №1, т.к. данный поток приоритетнее.

9.6 Завершение задачи при определенных условиях автоматически

В данный момент автоматическое завершение задачи не реализовано.

Возможны самостоятельные решения, рассмотрим на следующем примере:

Есть этап «Согласование СП». На выполнение задачи по этапу пользователю отводится 5 рабочих дней. За эти 5 дней пользователь может выполнить свою задачу с результатом либо «Согласовать», либо «Отклонить». Если прошло 5 дней и пользователь не завершил свою задачу, то она автоматически в фоне завершается с результатом «Завершено автоматически».

Для решения подобной задачи можно создать задание в Менеджере заданий в приложении Настройка системы, задать ему необходимое расписание и написать скрипт, который будет выполняться по данному расписанию. В этом скрипте необходимо написать запрос, который будет возвращать все подходящие задачи, написать получение идентификатора результата, с которым необходимо завершить задачу, и для каждой из задач вызвать метод

```
Bpm_TaskApi.setTaskResult(ropTask, idvResult);
```

который принимает в качестве параметров rop задачи ropTask и идентификатор результата idvResult, с которым эту задачу необходимо выполнить.